Titre: Title:	Multi-Fidelity Design Optimization of Francis Turbine Runner Blades
Auteur: Author:	Salman Bahrami
Date:	2015
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Bahrami, S. (2015). Multi-Fidelity Design Optimization of Francis Turbine Runner Blades [Ph.D. thesis, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/2027/

UNIVERSITÉ DE MONTRÉAL

MULTI-FIDELITY DESIGN OPTIMIZATION OF FRANCIS TURBINE RUNNER BLADES

SALMAN BAHRAMI

DÉPARTEMENT DE GÉNIE MÉCANIQUE

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION

DU DIPLÔME DE PHILOSOPHIAE DOCTOR

(GÉNIE MÉCANIQUE)

DÉCEMBRE 2015

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

MULTI-FIDELITY DESIGN OPTIMIZATION OF FRANCIS TURBINE RUNNER BLADES

présentée par : BAHRAMI Salman

en vue de l'obtention du diplôme de : Philosophiae Doctor

a été dûment acceptée par le jury d'examen constitué de :

M. VADEAN Aurélian, Doctorat, président

M. GUIBAULT François, Ph. D., membre et directeur de recherche

M. TRÉPANIER Jean-Yves, Ph. D., membre et codirecteur de recherche

M. LE DIGABEL Sébastien, Ph. D., membre

M. KOKKOLARAS Michaël, Ph. D., membre externe

DEDICATION

To my family

for their sincere love

ACKNOWLEDGEMENTS

This work could not have been succeeded to this extent without the support of my research director, colleagues, friends and family.

In full gratitude, I would like to express my special appreciation to my tremendous supervisor and research director, Professor Guibault, for his excellent guidance, caring, patience, availability and suggestions during my Ph.D. Also, I would like to express my special thanks to Dr. Tribes, who had a key role in my research project. I would like to thank him for his guidance and help. I also would like to thank the rest of the Polytechnique team and Dr. Devals, for his valuable help in CFD analyses and computational setups.

I greatly acknowledge Professor Vadean, Professor Trépanier, and Professor Le Digabel from Polytechnique Montréal, and Professor Kokkolaras from McGill University for their time devoted to evaluate my thesis.

I would also like to thank Andritz Hydro Canada Inc. for their helps and supports for this project and two internships, especially members of the R&D division: Mr. Desy, Mr. Vu, Mr. Nennemann, Mr. Murry, Mr. Gauthier, and generous assistance of Mr. von Fellenberg at the hydraulic design division.

The financial support of the NSERC for this research is acknowledged.

I take this opportunity to extend my sincerest thanks to all of my colleagues and teammates at Polytechnique Montréal for their valuable technical and emotional supports during all these years.

A special thanks to my wife; words cannot express how grateful I am for her continuous support and encouragement.

RÉSUMÉ

Ce projet de thèse propose une méthodologie Multi-Fidelity Design Optimisation (MFDO) qui vise à améliorer l'efficacité du processus de conception en génie mécanique. Cette méthodologie a été développée pour résoudre les problèmes liés à la conception mécanique des roues de turbines hydrauliques. Cette méthode peut être utilisée dans d'autres processus d'optimisation d'ingénierie, surtout si les processus d'optimisation sont coûteux. L'approche MFDO divise le coût informatique entre deux phases, une basse fidélité et une haute-fidélité. Cette méthode permet d'intégrer les avantages des évaluations à basse fidélité et haute-fidélité, et pour équilibrer le coût et la précision requise par chaque niveau de fidélité. Alors que la phase de basse fidélité contient la boucle itérative d'optimisation, la phase haute-fidélité évalue les candidats de conceptions prometteuses et calibre l'optimisation basse fidélité. La nouvelle approche de MFDO propose un Territorial-Based Filtering Algorithm (TBFA) qui relie les deux niveaux de fidélité. Cette méthode traite le problème que l'objectif d'optimisation à basse fidélité est différent de celui de la phase à haute-fidélité. Ce problème est commun dans les optimisations de substitutions basées sur la physique (par exemple en utilisant une analyse d'écoulement non visqueux à la place des évaluations d'écoulement visqueux). En fait, la vraie fonction n'est pas évaluable dans la phase basse fidélité due à l'absence de la physique impliquée dans ces évaluations. Par conséquent, les solutions dominantes de l'optimisation basse fidélité ne sont pas nécessairement dominantes du point de vue du véritable objectif. Par conséquent, le TBFA a été développé pour sélectionner un nombre donné de candidats prometteurs, qui sont dominants dans leurs propres territoires et qui sont assez différents du point de vue géométrique. Tandis que les objectifs de la phase haute-fidélité ne peuvent être évalués directement dans la phase basse-fidélité, certains objectifs peuvent être sélectionnés par des concepteurs chevronnés parmi des caractéristiques de conception, qui sont évaluables et suffisamment bien prédites par les analyses de basse fidélité. Des concepteurs expérimentés sont habitués à associer des objectifs de bas niveau à des bonnes conceptions.

Un grand nombre d'études de cas ont été réalisées dans ce projet pour évaluer les capacités de la méthodologie MFDO proposée. Pour couvrir les différents types de roues de turbines Francis, trois roues différentes ont été choisies. Chacune d'elles avait ses propres défis de conception, qui devaient être pris en charge. Par conséquent, différentes formulations de problèmes d'optimisation ont été étudiées pour trouver la plus appropriée pour chaque problème en main. Ces formulations ont exigé des configurations d'optimisation différentes construites à partir des choix appropriés de

fonctions objectif, les contraintes, les variables de conception, et d'autres fonctions d'optimisation telles que les budgets d'exploration locale ou globale.

ABSTRACT

This PhD project proposes a Multi-Fidelity Design Optimization (MFDO) methodology that aims to improve the design process efficiency. This methodology has been developed to tackle hydraulic turbine runner design problems, but it can be employed in other engineering optimizations, which have costly computational design processes. The MFDO approach splits the computational burden between low- and high-fidelity phases to integrate benefits of low- and high-fidelity evaluations, and to balance the cost and accuracy required by each level of fidelity. While the low-fidelity phase contains the iterative optimization loop, the high-fidelity phase evaluates promising design candidates and calibrates the low-fidelity optimization. The new MFDO approach proposes a flexible Territorial-Based Filtering Algorithm (TBFA) that connects the two levels of fidelity. This methodology addresses the problem that the low-fidelity optimization objective is different from the one in the high-fidelity phase. This problem is common in physics-based surrogate optimizations (e.g. using inviscid flow analyses instead of viscous flow evaluations). In fact, the real objective function is not assessable in the low-fidelity phase due to the lack of physics involved in the low-fidelity evaluations. Therefore, the dominant solutions of the low-fidelity optimization are not necessarily dominant from the real objective perspective. Hence, the TBFA has been developed to select a given number of promising candidates, which are dominant in their own territories and geometrically different enough. While high-fidelity objectives cannot be directly evaluated in the low-fidelity phase, some targets can be set by experienced designers for a subset of the design characteristics, which are assessable and sufficiently well predicted by low-fidelity analyses. The designers are accustomed to informally map good low-level targets to overall satisfying designs.

A large number of case studies were performed in this project to evaluate the proposed MFDO capabilities. To cover different types of Francis turbine runners, three different runners were chosen. Each of them had its own special design challenges, which needed to be taken care of. Therefore, variant optimization problem formulations were investigated to find the most suitable for each problem at hand. Those formulations involved different optimization configurations built up from proper choices of objective functions, constraints, design variables, and other optimization features such as local or global exploration budgets and their portions of the overall computational resources.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

1.1 Hydropower

Due to the depletion of non-renewable fossil energy, the enforcement of the Brazil-Kyoto agreements, and considering the high risk associated with nuclear power plants (especially after Fukushima Daiichi nuclear disaster in 2011), renewable energy markets are projected to continue to grow strongly in the following decades. Among all kinds of sustainable energies, hydropower persists to stand as one of the most important and reliable sources to meet the increasing energy demand. Figure 1-1 shows the portion of renewable energy and hydroelectricity among all energy sources. The global installed hydropower capacity increased from 715 GW to around 1000 GW between 2004 and the end of 2013 [1]. Hydro power had the minimum operation and maintenance labor cost among renewable energy technologies last year (see Figure 1-2).

The global hydroelectric output has always increased. It grew by 2.0% last year, which is below the 10-year average of 3.3% [2]. In spite of this growth, there is a big global potential as well. Norway and Paraguay produce almost all of their electricity (more than 98%) from hydropower resources [3]. Canada constitutes the third-largest generator of hydroelectricity in the world, despite a much smaller population than other key hydro players, China and Brazil. Canada had 77.6 GW installed hydropower capacity at the end of 2014, which accounts for 63% of the country's power generation. Surprisingly, there is a technical potential of adding 160 GW as well. Having already deployed 38.4 GW, Québec is the fourth-largest producer of hydroelectricity in the world after China, Brazil and the United States, and the largest producer in Canada [4].

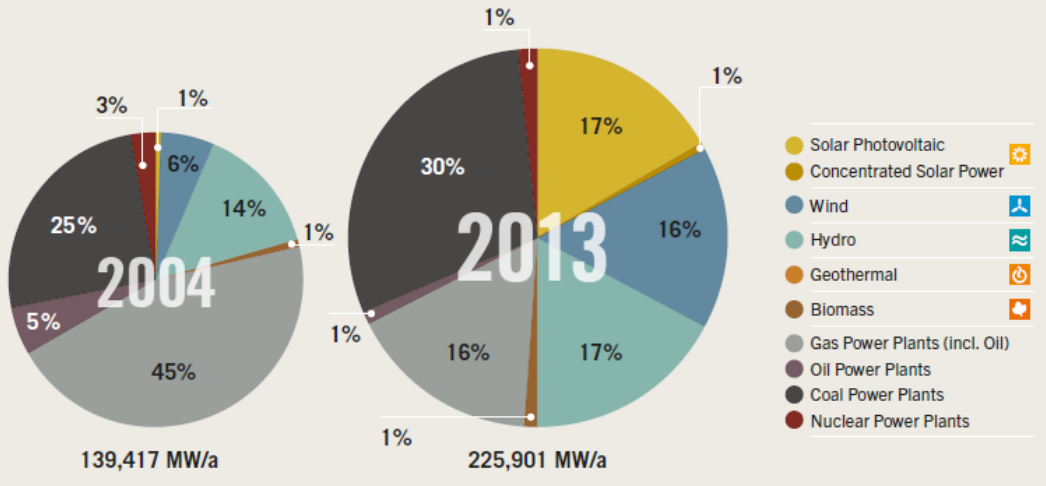


Figure 1-1 : Worldwide renewable energy market shares [1]

	Construction Times	Construction + Installation	Manufacturing	Operation + Maintenance	Fuel Supply
	Years	Job years / MW	Job years / MW	Jobs / MW	Jobs / PJ
Hydro Power	2	6.0	1.5	0.1	
Wind onshore	2	2.5	6.1	0.2	
Wind offshore	4	7.1	10.7	0.2	
Solar PV	1	9.0	11.0	0.2	
Geothermal	2	6.8	3.9	0.4	
Solar thermal	2	5.3	4.0	0.4	
Ocean	2	9.0	1.0	0.3	
Geothermal – heat		6.9			
Solar – heat		7.4			
Biomass	2	14.0	2.9	1.5	32.2
Biomass CHP		15.5	2.9	1.5	32.3

Figure 1-2 : Employment factors by renewable energy technologies at the end of 2014 [1]

1.2 Francis turbine

With more than 60% of the global hydroelectric generation, the Francis turbine is the most widely utilized type of turbine in the world. It is also the most commonly used turbine in Canada and Hydro-Québec's power systems.

James B. Francis developed the Francis turbine in 1848. As a reaction turbine, the pressure of the fluid changes as it passes through the immersed rotor blades. This type of turbine is quite versatile and adaptable to different projects, since its power output ranges from just a few kilowatts up to 1000 MW (see Figure 1-3).

Figure 1-4 shows the main components of a Francis turbine. The spiral casing receives the water, which is transferred from the lake behind the dam via the penstock. The spiral shape of the casing converts the axial flow into radial flow and distributes uniformly the flow into the stay vanes. The guide vanes are simultaneously adjustable and control the inflow characteristics by changing the inlet area (i.e. opening angle) and the angle of attack. The runner is connected to the generator by the shaft axis. It extracts the flow energy and converts the angular momentum of the flow into mechanical momentum. The draft tube recovers most of the flow kinetic energy by converting it into potential energy, which increases the effective head of the turbine.

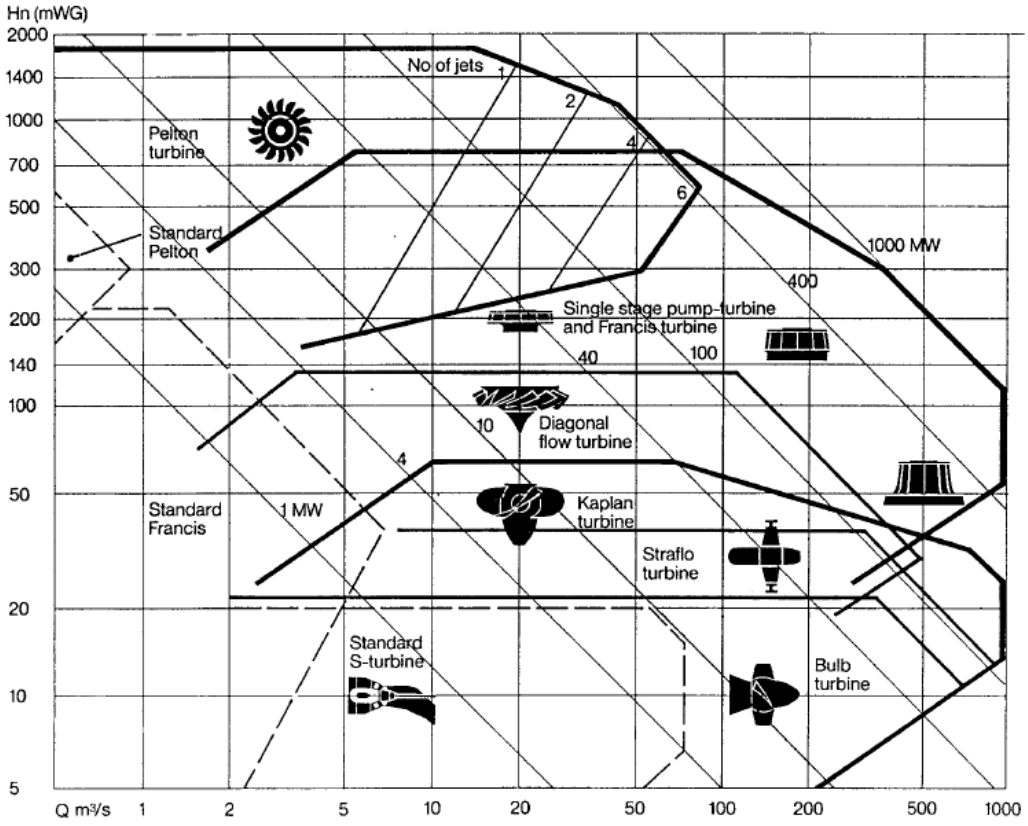


Figure 1-3 : Hydraulic turbine application range [5]

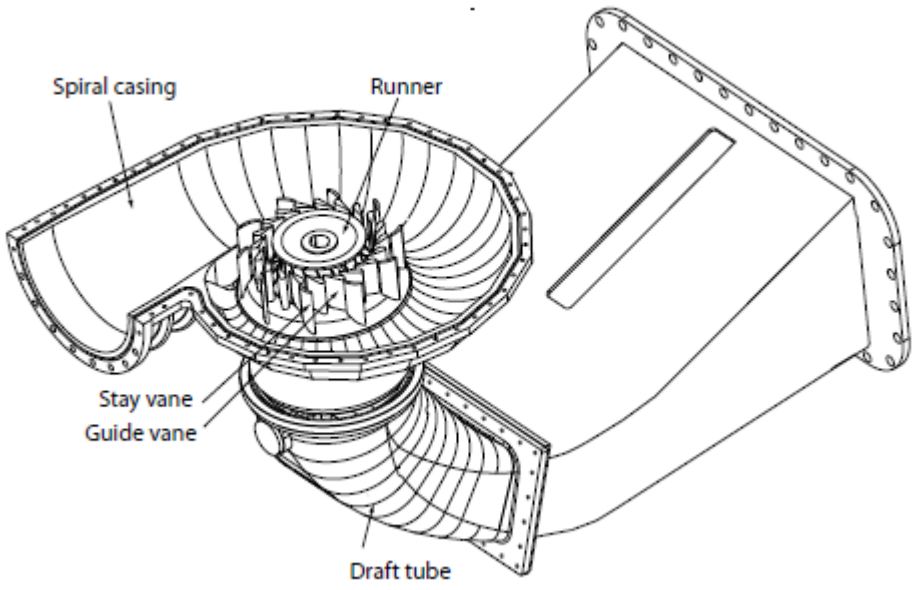


Figure 1-4 : Francis turbine [6]

1.3 Current hydraulic turbine runner design process

In a hydraulic turbine design process, designing the runner is one of the most challenging steps. Runner design has a huge influence on the design of other components. Also, the design of other components, particularly guide vanes and draft tube, affects the runner design. These design interactions are part of the runner design complexity, especially under the consideration of design time limits. The main challenges come from the nature of runner flows. Each runner is unique, since each design project has its own design criteria. To overcome these challenges, the designer needs a solid knowledge of fluid mechanics, as well as a deep understanding and excellent visual imagination of flow behavior inside the runner. He has to take into account different working scenarios and various operating conditions. After taking care of hydraulic design criteria, the designer should consider other disciplines such as structural and manufacturing criteria.

In hydrodynamic design, the designer employs all available evaluation tools, from the cheapest low-fidelity solvers to expensive high-fidelity CFD analyses. Figure 1-5 shows how the hydraulic design network connects the designer to the existing tools. While low-fidelity CFD analyses carry out the major portion of design iterations, high-fidelity CFD solvers evaluate the promising designs mainly in order to verify the main design characteristics and justify final tuning. The designer also uses expensive experimental investigations mostly to validate the final design. In addition, experimental investigations are performed to evaluate very complex phenomena that are difficult to predict with regular CFD tools, such as cavitation.

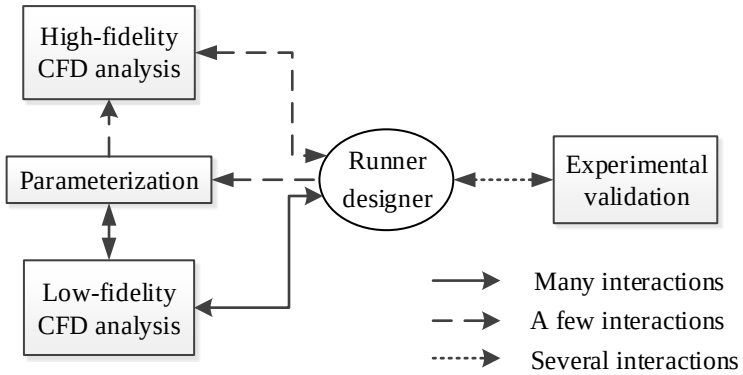


Figure 1-5 : Common industrial runner design process

Different designers may utilize different parameterization methods in order to play with geometric parameters. They usually spend a lot of time to manipulate the blade curvature and leading and

trailing edge shapes. Trailing edge shape is usually modified by changing the blade length on each blade section. Blade thickness profile is usually corrected later. Inner and outer contours are modified by changing the coordinates of several control points. For example, Figure 1-6 shows changing outer (band-side) contour by playing with cylindrical coordinates of two points on the contour. Also reducing the blade section length produces a shorter blade than the original one (represented by dashed line) by moving the trailing edge towards the upstream.

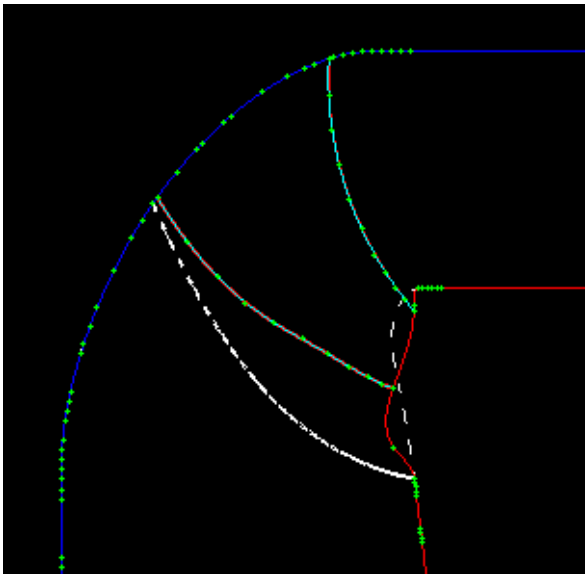


Figure 1-6: Modification of outer contour and trailing edge positions

The aforementioned points indicate that the runner designer's experience and intuition play a big role to accomplish this difficult task. This severe dependency causes a lot of designer interactions, which is a drawback considering the duration of the process. Also, the new design concepts may be trapped in a comfort design zone, which is inevitably built by past designer's experience. In the super-competitive global market, time and cost of the design process are always big concerns. Based on the aforementioned drawbacks of the current design process, these concerns are not properly taken into account in the current design framework.

1.4 Research questions and objectives

The present thesis proposes answers to the following main research questions:

1. How to reduce designer interactions in order to increase the efficiency of the runner design process?

2. How to integrate multi-fidelity runner flow analyses into an automatic design optimization methodology?
3. In the sake of runner performance improvement, what are the objectives and constraints, employed in which optimization configuration?

The aim of this Ph.D. research is to develop a new design methodology that optimizes the geometry of Francis turbine blades, in order to balance time and cost of design, and yield efficient runners, while adhering to different types of design constraints. To reach an efficient optimization methodology, a multi-fidelity framework has been developed, which can take maximum advantage of the low-fidelity model speed and high-fidelity model accuracy.

To effectively develop a design optimization tool, which is practical and reliable for designers, these specific objectives are defined:

1. Develop a multi-fidelity optimization methodology that relies on available industrial resources, e.g. parameterization methods and low- and high-fidelity flow solvers.
2. Investigate different low-fidelity optimization formulations to achieve the expected runner hydrodynamic performance improvement.
3. Demonstrate successful implementation of the proposed methodology and validate it by applying it to different Francis runner optimization problems.

1.5 Thesis overview and work organization

This thesis contains six chapters. Following the above introductory chapter, Chapter 2 provides a literature review of hydraulic turbine evaluation methods, different optimization categories, and surrogate-based multi-fidelity techniques. This chapter also covers the recent progress in hydraulic turbine evaluation and optimization.

Three articles will be presented in Chapters 3 to 5 respectively. They contain all materials presented in the three articles in the thesis format. Although there are some mutual materials in these three articles, they mainly focus on answering the following questions respectively:

1. What is the proper design optimization methodology considering the available runner flow analyses and parameterization tools?
2. In a multi-fidelity optimization methodology, what is the proper derivative-free optimization strategy and what are important optimization features based on characteristics and challenges of the problem at hand?

3. What are the characteristics of an infill method to connect low- and high-fidelity phases together?

Chapter 3 includes the first article entitled “Multi-fidelity shape optimization of hydraulic turbine runner blades using a multi-objective mesh adaptive direct search algorithm”, accepted for publication in the Elsevier journal of Applied Mathematical Modelling. This article proposes the newly developed multi-fidelity design optimization methodology and its components, formulations, and functionality validation through a medium-head Francis turbine runner optimization. This case study also investigates the effect of the number of overall loop iterations with fixed high- and low-fidelity computational budgets. The number of loops determines how often the low-fidelity problem formulation would be corrected as well. In this test case, the BIMADS algorithm handles two velocity objective functions with four constraints. Using two types of design variables represented by up to 17 parameters provides the lowest parameterization flexibility among all test cases.

Chapter 4 presents the second article entitled “Physics-based surrogate optimization of Francis turbine runner blades, employing mesh adaptive direct search and evolutionary algorithms”, published in the International Journal of Fluid Machinery and Systems. This article concentrates on evaluation of different optimization techniques using the MADS and an evolutionary algorithm applied to a low-head Francis runner optimization problem. Since this problem is more challenging than the first one, a new problem formulation is defined using a new objective function and three constraints. In addition, different optimization features are implemented to reach the feasibility and achieve the expected objective improvement. This article indicates the global and local capabilities of two aforementioned optimization algorithms and their advantages for the problem at hand.

Chapter 5 contains the third article entitled “Application of a territorial-based filtering algorithm in turbomachinery blade design optimization”, submitted in the Taylor & Francis journal of Engineering Optimization. This article explains the details of the proposed filtering algorithm, which is used as an infill method to choose new design points for costly high-fidelity evaluations. Two different blades are employed that belong to a hydraulic turbine and a transonic fan, in order to demonstrate the method functionality and the influence of important parameters. These two optimization problems utilize different optimization approaches. However, using different

objective functions in two design optimization levels allows us to apply the newly developed filtering algorithm in both problems and investigate its performance.

Among a large number of case studies performed in this project in the field of hydraulic turbines, only several cases have been selected to present in the aforementioned articles. In each article, a completely different Francis runner with different operating conditions is employed. Also various optimization formulations are used with different objective functions and constraints. Beside dissimilar optimization configurations, various design variables with different strategies are used in each article as well.

The sequence of articles shows part of the evolution of runner blade problem formulation. While the first case study presented in the first article applies tangential and meridional velocity objective functions, the second case study only considers tangential velocity as a constraint, and meridional velocity is completely removed from the optimization formulation. Instead, blade average length is taken into account as a new objective. It becomes doable by giving more flexibility to the geometry using more design variables dedicated to the blade length. In addition, three blade loading constraints employed in the first article are replaced with a new one in the next articles.

Chapter 6 provides a general discussion about the results illustrated in the articles and describes briefly their connections. This chapter also demonstrates how the articles complete each other to demonstrate the functionality and performance of the proposed optimization methodology. Finally, Chapter 7 gives the work conclusion and contributions followed by several recommendations for future works.

CHAPTER 2 LITERATURE REVIEW

This chapter only presents the literature most relevant to the research at hand. The first section explains some choices of hydraulic turbine performance evaluations. They can be employed in the optimization methods described in the second section. The third section introduces some surrogate-based optimization methods that can use multi-fidelity evaluations. The fourth section summarizes the chapter.

2.1 Hydraulic turbine performance analyses

In this section several methods of hydraulic turbine flow field analysis are briefly described. It includes the description of several methods ranging from the highest fidelity methods to the lowest ones, with their application cases.

2.1.1 Experimental analysis

Experimental analysis of a hydro turbine is the most accurate methods of flow field investigation that designers use mostly at the final design step. They also use experimental investigations to validate CFD tools. Experimental analysis includes the minimum assumption and the maximum detail of the phenomena. There are some experimental studies of Francis turbines recently cited by researchers are presented below.

Wang et al. [7] studied the unsteady behavior of a prototype 700 MW Francis turbine unit for a 200–700 MW load range with water head of 57m to 90m. They concluded that pressure fluctuations in the draft tube are always stronger than that of upstream flow passages.

Susan-Resiga and Muntean [8] studied the flow characteristics at the outlet of a Francis turbine runner experimentally to investigate the causes of a sudden drop in the draft tube pressure recovery coefficient at a discharge. They used Laser Doppler Velocimetry (LDV) measurements to determine both axial and circumferential velocity components at the runner outlet.

Ciocan et al. [9] analyzed LDV flow survey, pressure and wall friction measurements at the runner outlet using phase average techniques. They aimed to investigate the influence of the rotation and passage of the blade wakes. They also characterized the turbulence and pressure fluctuations.

Tridon et al. [10] focused on the radial velocity components of the swirling flow in a Francis turbine draft tube. Velocity measurements were carried out at CREMHyG (Grenoble) using LDV and PIV[1] techniques at four operating points.

The FLINDT project was carried out for a better understanding of flow physics in Francis turbines and to create an extensive experimental database describing a wide range of operating points, which can provide a firm basis for the evaluation of the CFD engineering practice. ALSTOM, Electricite de France, EPFL, General Electric Canada, Va Tech Hydro and Voith Siemens can be mentioned as the main partners of this project. The experimental data are available in several references such as Ref. [11].

Experimental studies are an important part of the designer's toolbox that can be considered as the highest level of fidelity for turbine performance prediction. However, they cannot be used during the iterative design optimization mainly due to the high cost and time required for these studies.

2.1.2 Mathematical analysis

Due to some simplifying assumptions in the construction of mathematical models, they usually have been categorized as low or medium fidelity models. The main idea is that instead of computing the flow in a certain domain, some algebraic equations can be applied based on operating conditions and kinematic constraints. Thus, runner designers sometimes employ mathematical models to calculate velocity components at a certain operating condition with acceptable accuracy.

Wang and Rusak [12] studied vortex breakdown, and provided a theoretical understanding of swirl flow dynamics. They developed a mathematical model based on the axisymmetric Euler flow. They considered theoretical swirl flow configurations dedicated to a one parameter Batchelor vortex. Leclaire and Sipp [13] did the same investigation with a two-parameter vortex.

Susan-Resiga and co-workers [14] have developed a mathematical model of the swirling flow in Francis turbines for a wide range of operations. They assumed an inviscid steady swirling flow with vanishing radial velocity at the runner outlet. They investigated the correlation between the moment flux of momentum downstream the runner and the operating regime given by the turbine

[1] Particle Image Velocimetry, which allows making non-intrusive fluid flow velocity measurements.

discharge and head. They represented the relationship between the axial and circumferential velocity components using a swirl-free velocity instead of the traditional relative flow angle at the runner outlet. It has been shown that the swirl-free velocity approach is more suitable to describe the swirl kinematic at the runner outlet. This concept was employed by Kubota et al. [15] to investigate draft tube losses. Kubota and co-workers used a single value corresponding to an arbitrary chosen streamline, and did not consider the axial and circumferential velocity profiles. The swirl-free velocity can be written as:

$$V_{sf} = \Omega R V_{2Z} / (\Omega R - V_{2\Theta}) \qquad (2\text{-}1)$$

where V_{2Z} and $V_{2\Theta}$ are axial and circumferential velocities at S_2 and Ω is the runner angular speed. R is the radius measured from the runner axis. Figure 2-1 shows the velocity triangle at the runner downstream (S_2 in Figure 2-2). β_2 is the relative flow angle, which was assumed in this research to depend only on the radius for the section S_2.

The main advantage of Susan-Resiga's model is its ability to compute radial profiles of axial and circumferential velocity components at the runner outlet, without any computation of flow in the turbine. As they have validated the proposed mathematical methodology, it can be used as a low-fidelity solver. By achieving an optimum swirling flow configuration at the runner outlet, the shape of runner blades can be reached through an inverse design approach.

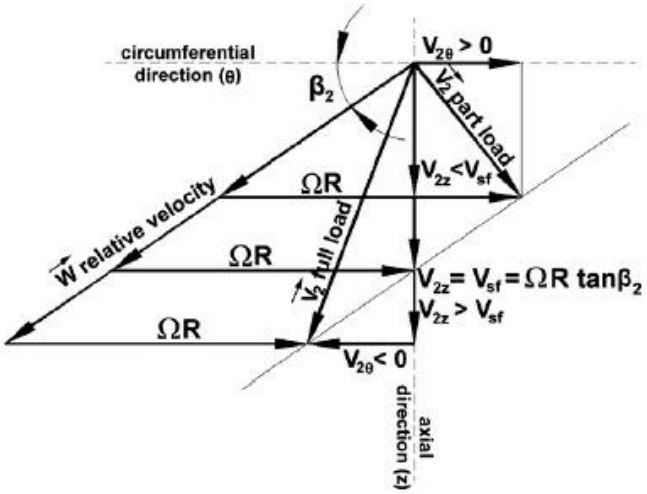


Figure 2-1 : Velocity triangle downstream the runner [14]

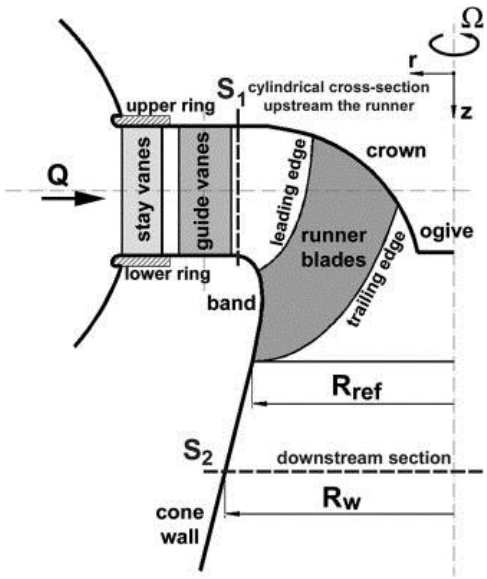


Figure 2-2 : R-Z cross section of a Francis runner [14]

2.1.3 CFD analysis

Since a few decades ago, the table-look up and inflow model approach and expensive experimental studies are being complemented or replaced by Computational Fluid Dynamics (CFD) analyses. CFD tools are usually used to meet the following objectives [16]:

- Understanding details of complex flow behaviors such as cavitation, unsteady vortex shedding, and flow instabilities, which need special modeling due to the phenomenon nature.
- Prediction of hydro turbine performance at design and off-design operating conditions.
- Avoiding excessive investigations in the lab and on-site.
- Cost-efficient iterative evaluations required during optimization processes.

Runner designers widely use various CFD analyses to study the flow and evaluate the candidates in different steps of the design process. Basically, CFD tools employ one of the following models:

I) Inviscid flow models:

Inviscid flow models assume an ideal fluid which has no viscosity. Although no practical flow is inviscid, the inviscid flow assumption can be applied when viscous effects can be neglected. For instance, very high Reynolds-number flows such as high-speed external flows around streamlined

bodies (far enough from walls) can be treated as inviscid flows. Since the Reynolds number represents a ratio of convective to diffusive influences, a very high Reynolds number indicates negligible diffusion. Also, the inviscid flow assumption is valid when time scales for diffusion are much larger compared to the time scales for convection. In 2D, the governing equations of such flows are known as the 2D Euler equations:

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} = -\frac{1}{\rho}\left(\frac{\partial p}{\partial x}\right) \quad (2\text{-}2)$$

$$\frac{\partial V}{\partial t} + U\frac{\partial V}{\partial x} + V\frac{\partial V}{\partial y} = -\frac{1}{\rho}\left(\frac{\partial p}{\partial y}\right) \quad (2\text{-}3)$$

The continuity is integrated in the momentum equation. In these equations, *x* and *y* define the Cartesian position, *U* and *V* are Cartesian velocity components, *p* is the static pressure, and ρ is the fluid density.

Potential flow analysis as a low-fidelity method has been applied widely to optimize the shape of hydraulic turbine components. In fact, it is an irrotational Euler flow. One of the best explanations of the potential flow theory is still Holmes and McNabb's paper [17]. They developed a package of computer programs for the flow analysis through hydraulic turbines. The solution algorithm is a fully three-dimensional Galerkin finite element analysis, using a pre-conditioned conjugate gradient equation solver. Holmes and McNabb applied their program to a Francis turbine runner design to reduce the cavitation problem. They verified the proposed changes of runner in reducing the cavitation and efficiency enhancement in scale models tests and on the full-scale runner.

Inviscid analyses can assist the designers to approach quickly the desired targets with very low cost of computation and time. Also, it is one of the most commonly used techniques for the preliminary design step of hydraulic turbine components. For instance, Wu et al. [18] applied inviscid quasi-3D and 3D Euler codes in the preliminary stage of runner optimization.

II) Viscous flow models:

The Navier–Stokes equations describe the motion of viscous fluids. The numerical solution for a turbulent flow is difficult. A proper turbulence model can be chosen based on some criteria that are specific to each problem, such as required physics, time limit and computational resources. Figure 2-3 lists several turbulence models sorted from the simplest to the most complex [19]. Among them, several Reynolds Averaged Navier-Stokes (RANS) models are more common in

hydro turbine applications, which are adaptable and efficient to calculate flow fields. One case of successful applications for each of those models is presented in this part.

RANS models solve Reynolds averaged Navier-Stokes equations (Eq. 2-5), which are based on the decomposition of the quantities into mean and fluctuating terms. For instance, Eq. 2-4 shows the decomposition of the velocity into average (U_i) and fluctuating terms ($u_i(t)$). Time-averaged statistics of turbulent velocity fluctuations are modeled using functions containing empirical constants and information about the mean flow. RANS models require closure for Reynolds stresses, which is shown in Eq. 2-6. In this equation, μ_t is the turbulent viscosity. For the k-ε model it is calculated from Eq. 2-7, where C_μ is a constant turbulent quantity. For instance, in the k-ε models, turbulent viscosity is correlated with turbulent kinetic energy, k, and its dissipation rate (i.e. ε) [19].

$$U_i(t) \equiv U_i + u_i(t) \tag{2-4}$$

$$\rho U_k \frac{\partial U_i}{\partial x_k} = -\frac{\partial p}{\partial x_i} + \mu \frac{\partial^2 U_i}{\partial x_j \partial x_j} + \frac{\partial R_{ij}}{\partial x_j} \tag{2-5}$$

$$R_{ij} = -\rho \overline{u_i u_j} = -\rho \frac{2}{3} k \delta_{ij} + \mu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \tag{2-6}$$

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{2-7}$$

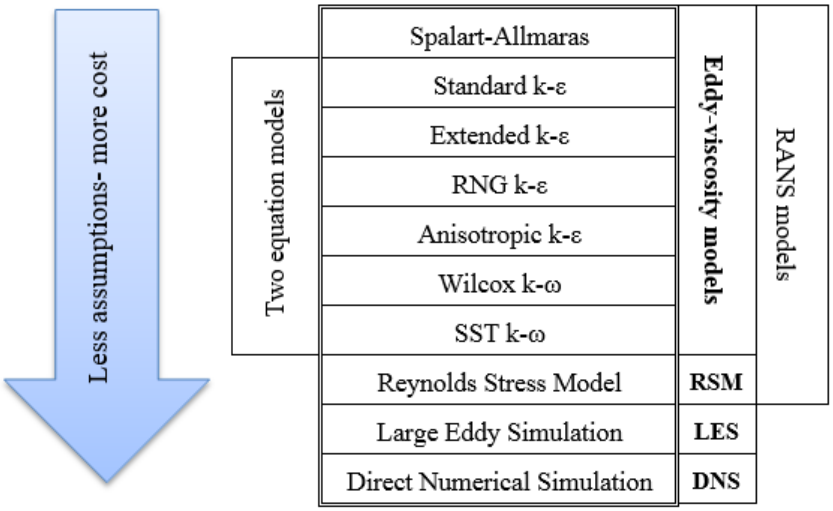


Figure 2-3 : Turbulence models

Wu et al. [18] applied a CFD-based design system to a Francis turbine runner and a tandem cascade. They used a standard k-ε turbulent flow solver, with special attention paid to flow matching between stationary and rotating parts. Wu and co-workers concluded that 15 blades is an optimal number, due to the compromise between increasing friction loss and flow blockage, and decrease of overall pressure loading per blade by increasing the number of blades. They observed a vortex generated on the pressure side near the leading edge, but it disappeared near the band. The authors indicated that it could be a result of weaknesses of the standard k-ε model and wall function to predict swirling and separated flows.

Franco-Nava et al. [20] studied a Francis turbine runner numerically and optimized it based on the genetic algorithm. The Spalart-Allmaras model was used as the turbulence model. To define appropriate inlet flow conditions of the runner, CFD analysis for the wicket gate was carried out too. At the runner outlet, average static pressure was applied based on the experimental data. The authors concluded that the manufacturing and mechanical integrity should be considered in the runner optimization as well.

Hu et al. [21] applied RANS simulations to study the unsteady turbulent flow through Francis turbine components. They used the Renormalization Group (RNG) k-ε turbulence model, which was modified in 1986 based on the theory of fuzzy mathematics in order to take into account higher accuracy of prediction of swirling flow influences [22]. Hu and co-workers simulated all components starting from inlet of the spiral casing and ending at the draft tube outlet, using finite volume commercial software, CFX-TASC-flow. Two slip-surfaces were applied between rotating and stationary components, and a mixing surface was applied at the mid-face between the outlet of the guide vanes and the runner inlet.

Yaras and Grosvenor [23] carried out general studies on the axisymmetric separating and swirling flows numerically and experimentally. They evaluated five turbulence models: the k-ε model of Chien [24], the two-layer k-ε model of Rodi [25], the k-ω model of Wilcox [26], the two-equation Shear-Stress-Transport (SST) model of Menter [27], and the one-equation eddy-viscosity model of Spalart and Allmaras [28]. None of the models used wall-function boundary conditions. They concluded that all models (except Chien's k-ε model) were successful in capturing the surface pressure and skin friction distributions in an axisymmetric separating flow. Also in all cases, a slight over-prediction of static pressure in the separated zone was observed. Menter's SST model

was the most successful in capturing the velocity profiles, and Rodi's k-ε model was the weakest one in this respect. All models failed to predict the peak k value in the boundary layer. In terms of minimum grid resolution for acceptable prediction accuracy for boundary layer, Rodi's k-ε and Spalart-Allmaras models showed the best performance, requiring a maximum of Yplus=5 and at least 15 nodes within the boundary layer. One of the important conclusions of this research was that all those turbulence models overestimated significantly the radial diffusive transport in the case of strongly swirling confined flow, and SST model yielded the worst prediction.

Susan-Resiga et al. [29] analyzed numerically the swirling flow downstream a Francis runner. They used a simplified straight conical diffuser in order to focus on the decelerated axisymmetric swirling flow in the draft tube cone. They employed a Reynolds Stress Model (RSM) in the commercial code, FLUENT 6.2.16, with a nonequilibrium wall function. The RSM model involves calculation of the individual Reynolds stresses using differential transport equations. They are employed to obtain closure of the Reynolds-averaged momentum equation. Resiga and co-workers concluded that RSM with a quadratic pressure-strain term can predict accurately the flow behavior. They also investigated a flow control technique, which utilized a water jet injected from the runner crown tip along the axis to remove the vortex breakdown at partial load.

Susan-Resiga and co-workers [30] had used Realizable k-ε (RKE) before these investigations, to compute the circumferentially averaged swirling flow in the discharge cone of a Francis turbine at low discharge conditions. The RKE model was developed by Shih et al. [31] to provide superior performance for flows involving rotation, boundary layer under strong adverse pressure gradient, separation, and recirculation. This model satisfies certain mathematical constraints on the Reynolds stress, consistent with the physics of the flow.

Large Eddy Simulation (LES) needs a much finer mesh than what RANs-based models need, and is employed to simulate unsteady flow phenomena. For instance, Pacot and co-workers [32] applied LES to simulate rotating stall phenomena in partial load of a pump turbine machine using about 100 million hexahedral elements. Due to the high computational cost, LES has been recently employed in hybrid RANS-LES turbulence models. For instant, Krappel et al. [33] carried out Francis pump turbine flow simulations at part load conditions using a SST-LES turbulent model with two grids containing about 10 and 20 million cells for the whole machine.

2.2 Optimization methods

Optimization methods can be classified into several categories depending on the type of problems and their specifications, such as continuous and discrete, global and local, linear and nonlinear, single-objective and multi-objective optimizations. Also, two categories of optimization algorithm can be distinguished: the gradient-based and non-gradient (derivative-free) methods, which affect strongly the possibility, robustness, required information for the optimization process, cost of optimization, and the solution quality. In engineering optimizations, almost all problems are subject to constraints which divide the design space into feasible and infeasible regions. In this section, those categories that are more relevant to this research are briefly described.

Figure 2-4 shows the general flowchart of an optimization problem, which employs a high-fidelity CFD chain. This chain includes a mesh generator, a viscous flow solver, and a post-processing. Using a high-fidelity CFD solver lonely in turbomachinery shape optimizations is quite time consuming and computationally expensive. For instance, Flores et al. [34] optimized a Francis runner represented by 24 design variables. They employed a high-fidelity CFD flow solver using the SST turbulence model during the optimization. Flores and co-workers reported that they performed 1000 model evaluations; each model was discretized by about 300000 cells. Each computation took about 15 minutes by parallelizing over four processors. Therefore, the overall time was about 250 hours (i.e. more than 10 days).

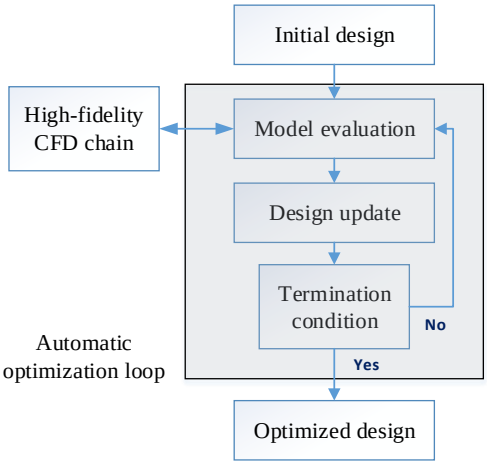


Figure 2-4 : Optimization flowchart

2.2.1 Multi-objective optimization

Modern engineering problems usually involve several design objectives and optimal solutions are found as a trade-off among them. In turbomachinery problems dedicated to shape optimization, designers should consider different design aspects (e.g. drag, lift and cavitation). Those objectives may be in conflict and the optimal parameter of one objective usually does not lead to optimality of other objectives. In fact, multi-objective optimization will lead to a set of solutions, called an approximation of Pareto optimal set or Pareto front, which are not dominated by other optimization solutions. Solutions can be chosen after employing additional decision-making criteria (see Figure 2-5).

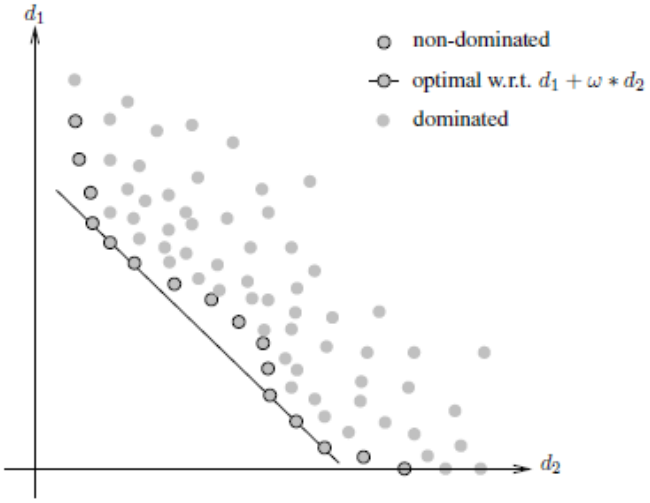


Figure 2-5 : Pareto front for two objective functions and a decision criterion example

2.2.2 Gradient-based methods

Gradient-based algorithms start from an initial point, and utilize gradient information to decide where to move. Steepest descent and conjugate gradient methods are first-order methods, which typically exhibit linear convergence due to their use of function gradients. The Newton method, as a second-order method, additionally employs the Hessian to reach the minimum. The gradient vector and Hessian matrix can be approximated using finite differences if they are not available analytically, or using adjoint and complex step methods. Figure 2-6 presents a cost comparison of these methods to calculate the gradient vector. In this figure, the time is normalized with respect to the time required for one solution of an aero-structural design [35].

For instance, Tatossian et al. [36] developed an aerodynamic shape optimization approach to improve the performance of hovering rotor blades represented by about 4000 design variables in transonic flow using a discrete adjoint method. Figure 2-7 shows the general design flowchart used in their work. For each design run, 500 multigrid cycles were employed for the flow and adjoint solvers, where each run required between 4 and 7 hours on 12 processors. They performed 25 design cycles to ensure that the minimum had been achieved.

Although gradient-based methods are known as powerful methods for local search, there are some challenges in real problems such as evaluation failures, prohibitive computational cost, and numerical noise that make difficult their systematic use.

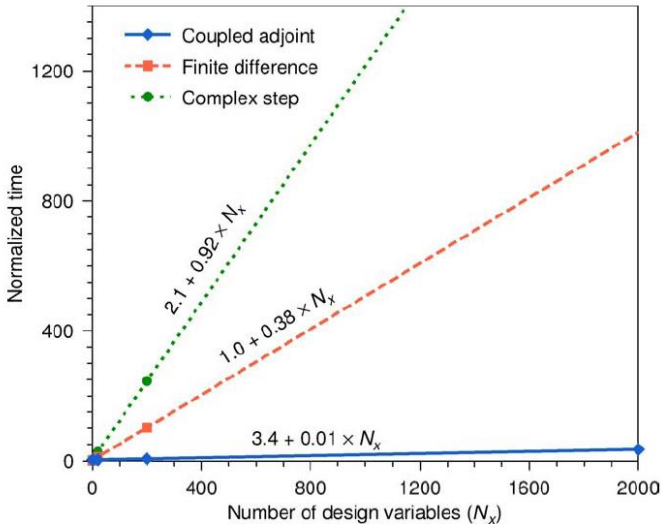


Figure 2-6 : Computational time comparison of gradient calculation [35]

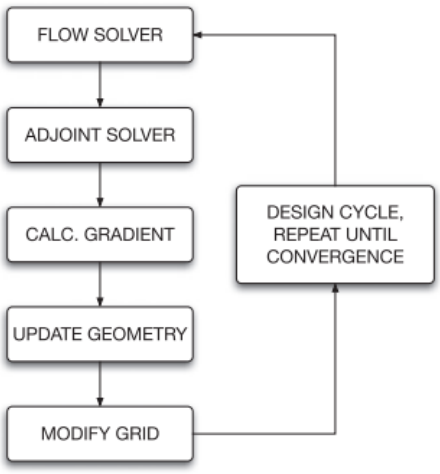


Figure 2-7 : Adjoint-based design flowchart [36]

2.2.3 Non-gradient-based methods

Many practical applications require the optimization of functions whose gradients are not available or computationally expensive or time consuming to compute. Therefore, various methods have been developed that use the function values at a set of sample points to determine new design points. They can be appropriate algorithms to find different local and global minima in non-convex and discontinuous objective functions as well as discrete spaces. They can also handle noisy objective functions. Non-gradient-based methods are appropriate choices where function evaluations are the results of computer codes, i.e. blackbox, which may fail even for feasible design points. On the other hand, these methods usually require a much larger number of evaluations than gradient-based methods.

Among derivative-free methods, direct-search algorithms (such as MADS [37], Pattern Search [38], and Generalized Pattern Search [39]) have some proofs of convergence to local optimality. In contrast, there are considerable numbers of derivative-free global search methods, such as genetic algorithms [40], evolutionary algorithms [41], particle swarm optimization [42], and simulated annealing [43]. They do not have any optimality convergence guarantee.

Most hydro blade shape optimizations have employed derivative-free optimization techniques [34, 44, 45]. From each aforementioned category (i.e. local and global search derivative-free methods), one optimization method is described in two next following subsections, which have been employed in the present research work.

2.2.4 Direct search algorithms

In direct search algorithms, a set of directions with suitable features are used to generate a finite set of points at which the objective function is evaluated [46]. One of the most recent direct search methods is the Mesh Adaptive Direct Search (MADS) developed by Audet and Dennis [47]. This method is used by NOMAD software employed in case studies presented in this thesis. The MADS algorithm iterates to evaluate blackbox functions at some trial points located on a mesh, to improve the current best solution. The mesh is a discretization of the design space. Figure 2-8 shows the MADS flowchart. Each iteration is composed of two steps: the Search and the Poll.

The Poll generates trial mesh points in certain directions in the vicinity of the best current solution and evaluates those points. When the iteration results in no improvement with respect to objectives and constraints, the next iteration will be initiated on a finer mesh. This algorithmic feature

provides the basis for the convergence analysis of the overall optimization process [48]. Figure 2-9 illustrates an unsuccessful Poll step with mesh refinement. In the first iteration, evaluation of three trial points has not caused function improvement and x_k has remained as the best current point. Therefore, new trial points with new directions are set on the refined mesh. The Poll size is directly related to the mesh size and is reduced slower than the mesh size. Therefore, more directions are provided for the new iteration.

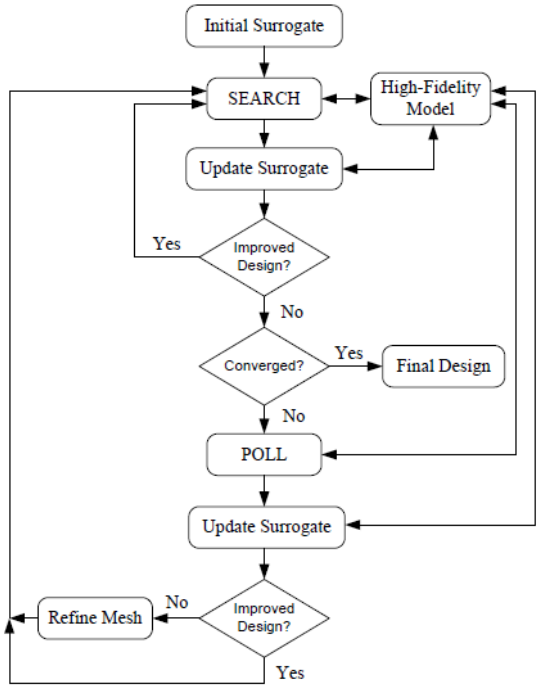


Figure 2-8 : MADS flowchart [49]

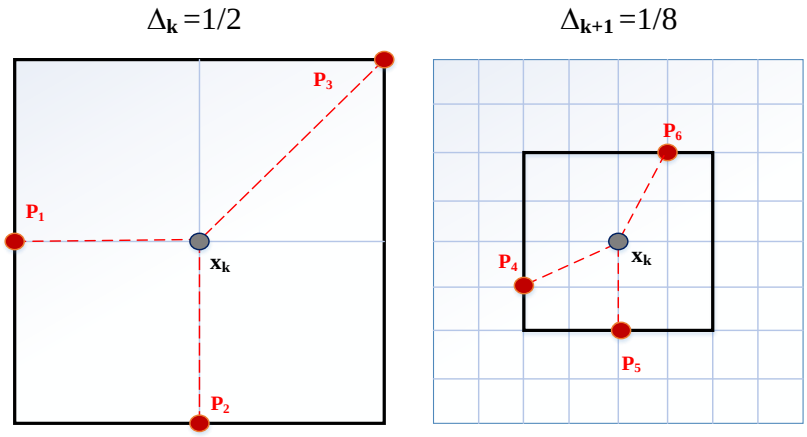


Figure 2-9 : Mesh shrinkage in a Poll step

The Search step is carried out before the Poll step, which can explore the design space using a surrogate model. The Search can return any point on the underlying mesh in order to improve the current best solution. The default Search step usually uses a quadratic model of all functions by using available evaluations and conducting an optimization on this model. In addition, a variable neighborhood search (VNS) [50] can be employed to escape from local minima.

The MADS algorithm treats constraints in three different ways. The first type of constraints, unrelaxable constraints, cannot be violated by any trial point. In other words, function evaluation will not be considered if at least one of these constraints is violated. For instance, design variable bounds can be treated as unrelaxable constraints. The second type, relaxable constraints, can be violated and the amount of violation is measured. The third type, hidden constraints, is mainly dedicated to blackbox evaluation failures.

The main convergence criterion of the MADS algorithm is met when the mesh size becomes small enough. However, in practice the maximum number of evaluations can be set to stop the optimization with respect to available computational budget. The BIMADS algorithm [51] has been developed to handle bi-objective optimization problems. It solves a series of single-objective sub-problems of the bi-objective problem using the MADS algorithm to obtain an approximation of Pareto front [52].

2.2.5 Evolutionary algorithm

Popular Evolutionary Algorithms (EAs) have been inspired by Darwin's theory of evolution. The EAs are random population-based methods, which are widely used in engineering design due to their robustness and ability to handle single- and multi-objective, constrained optimization problems without getting trapped in local minima [53]. They are also able to obtain Pareto front approximation even after the first run, since in each run all population is evaluated. EAs usually require a large number of function evaluations compared to gradient-based algorithms, which can be their main drawback.

Unlike the MADS algorithm that is relatively new, the first attempts of EAs usage are dated back to the 1950’s, and were performed by Friedberg [54], Bremermann [55], and Box [56]. They initiated the development of three different classes of EAs: evolutionary programming, evolutionary strategies, and genetic algorithms.

EAs handle populations of individuals representing potential solutions, evolving from generation to generation. Each individual is evaluated to calculate the fitness or cost value based on the objective function. Then the EA selects the most fitted individuals among the last generation, called parents, in order to evolve them using evolution operators, recombination (or crossover) and mutation (see Figure 2-10). The new population is called offsprings or children, which is expected to be more adaptable to the environment [57]. In the second article in Chapter 4, EASY employs the aforementioned EA algorithm.

The EAs usually consider constraints by applying penalty functions [58], which significantly decrease the chance to survive for individuals violating the constraints.

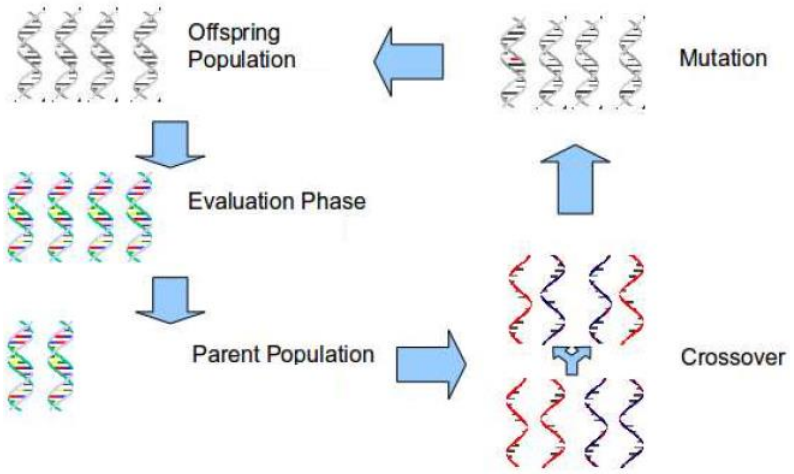


Figure 2-10 : Schematic of an EA loop [57]

2.3 Multi-fidelity surrogate-based optimization

High-fidelity investigations are quite accurate, but typically time consuming and expensive. In most industrial optimization problems, it is not preferable to use costly high-fidelity evaluations in the main iterative optimization loop, especially by using demanding derivative-free methods. Therefore, multi-fidelity optimization methodologies have been developed to integrate lower fidelity evaluations in conjunction with the high-fidelity ones, in order to combine their advantages and alleviate their drawbacks. Multi/variable fidelity optimization methods usually have different phases/levels. In the low phase, Surrogate-Based Optimization (SBO) techniques are mostly employed in order to decrease the number of high-fidelity evaluations. In such a framework, a greater quantity of relatively cheap information can be coupled with a small amount of expensive information to increase the accuracy of the surrogate model (if required or possible). It has been

shown that the multi-fidelity SBO methods are more scalable to larger numbers of design variables and much less high-fidelity evaluations are needed to obtain a given accuracy level [49].

Refs. [59-61] have provided comprehensive studies and overviews of different surrogate methods and their implementation in the literature. The surrogates can be created by applying each type of these methodologies, or a combination of them:

1. By using mathematical approximations of the high-fidelity model named functional surrogates.
2. By using a reduced-dimension space.
3. By using reduced physics (e.g. inviscid flow solvers for flow field calculation).

In the case of using CFD tools, two more choices will be added:

4. By using variable-resolution models, which means the same high-fidelity solver is used, but with a coarser grid.
5. By using variable-accuracy models, which means the convergence tolerance is reduced in high-fidelity CFD analyses.

Numbers 3 to 5 are also called physics-based surrogates [49].

2.3.1 Functional surrogates

Functional or mathematical surrogates (also called surrogate models) consist of approximation models that mimic the behavior of the simulation model. They are usually constructed based on modeling the response of the simulator to a number of data points, without any particular knowledge of the physical system. Functional surrogate optimization contains three main steps, which can be iteratively interleaved (see Figure 2-11):

1. Selection of initial data points, also called sequential or optimal experimental design
2. Surrogate model construction
3. Surrogate model optimization
4. Surrogate accuracy assessment

To enhance the surrogate model accuracy, some new points should be chosen and evaluated. They are known as infill points, which are described later in Subsection 2.3.4.

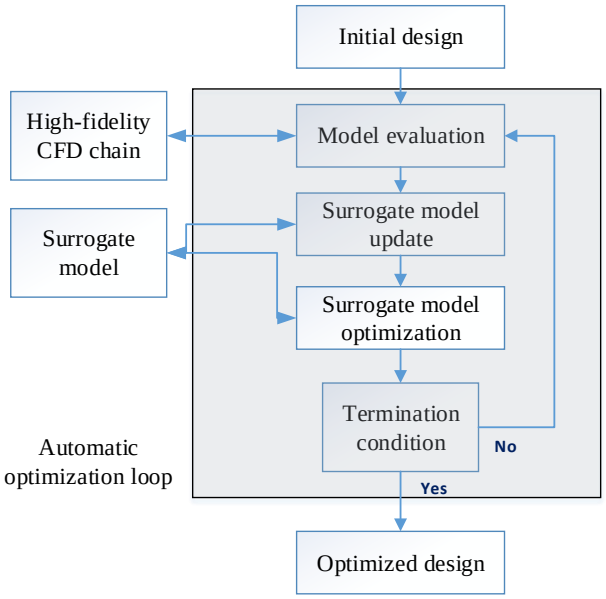


Figure 2-11 : Flowchart of a functional SBO algorithm

One of the most widely used forms of surrogate models is the Polynomial Response Surface Model (PRSM). A comprehensive overview of PRSM is presented in Ref [62]. The PRSM consists of a group of mathematical and statistical techniques utilized in the development of a relationship approximated by a low-degree polynomial model, between an interest response and some variables.

Radial Basis Functions (RBFs) use a weighted sum of radially symmetric functions to emulate complicated design landscapes. Typical basis functions are linear, cubic, and thin plate spline. Employing parametric basis functions can bring more flexibility (e.g. Gaussian) [60]. For instance, Georgopoulou et al. [63] optimized runner blades of Francis and Kaplan turbines using an RBF in a hierarchical surrogate-based evolutionary algorithm (EA). Earlier, they had shown the role of employing this method in multi-objective EA in mathematical and aerodynamic shape optimization problems [64]. The reported results of Francis runner design after 400 exact evaluations showed no significant improvement of cost functions in comparison with the results of 400 Euler-based evaluations. For Kaplan runner optimization, they carried out 3500 high-fidelity evaluations, which is quite high. It indicates that the developed surrogate-based multi-fidelity design methodology is not efficient enough due to its disability to employ all benefits of low-fidelity and high-fidelity models.

Kriging is another well-known surrogate model named in honor of the South African mining engineer Danie Krige, who developed and applied mathematical statistics in ore evaluations [60,

65]. It constitutes a Gaussian-based modeling method as a particular case of RBF models. Unlike most functional surrogates, Kriging does not assume independent error terms (i.e. residuals). It assumes a correlation between the residuals of two design points related to their spatial distance. Due to its expenses, it is usually used when the true function is computationally expensive, e.g. CFD-based calculation [60]. For instance, Jouhaud et al. [66] applied a Kriging model to optimize a 2D airfoil shape. A low-dimensional model parameterized the geometry, which consisted of two design variables.

The Artificial Neural Network (ANN) method is based on the neuron function. The network is trained by solving a nonlinear least-square regression problem for a set of training points. A typical ANN consists of several layers each including several nodes. The ANN receives the input data from the input layer using input nodes. Then, at least one intermediate or hidden layer contains hidden neurons that stand for computational units. Neuron connections transfer data between different layers. The last layer contains output nodes that deliver the final ANN response [59].

More recently, multi-surrogate techniques have been developed to employ more than one functional surrogates with the same evaluation points. This combination can eliminate the risk of wrong surrogate model selection and provides the robustness to choose the most proper one at each optimization level. For instance, Badhurshah and Samad [67] incorporated PRSM, RBF, and Kriging to optimize a bidirectional impulse turbine blade. The high-fidelity results obtained by RANS simulations were used to train the surrogates and find the optimal points via a hybrid genetic algorithm. They also applied a weighted average surrogate. While multi-surrogate usage was quite satisfactory, the weighted average surrogate did not have the expected performance. Badhurshah and Samad reported a high number of 600 iterations and an average time of 6 hours for a single simulation. Vesting and Bensow [68] employed several ANNs and Kriging models and a combination of them in a marine propeller optimization. They also reported a large number of iterations that were time consuming even with eight parallel computations.

2.3.2 Physics-based surrogates

In the case of a CFD-based optimization, physics-based surrogates use simplified governing equations, coarser discretization grids, or relaxed convergence tolerances, or a combination of them. Unlike functional surrogates, physics-based surrogates cannot be corrected and consequently

their accuracy cannot be improved. However, the optimization problem can be corrected to yield more global accuracy and to approach actual optimal solution points.

Jameson and co-workers employed an adjoint method in a SBO of airfoils, using simplified physics; potential flow [69] and compressible Euler flow [70]. Forrester et al. [71] used partially converged CFD results as physics-based surrogates to build a Kriging approximation. They concluded that partially converged results can produce globally more accurate surrogate models than converged simulations for a given computational budget. In addition, they reported a 48% time saving achieved in a three-dimensional wing problem. Leary et al. [72] also combined physics-based surrogate using coarse grids with functional surrogate using an ANN.

In hydraulic turbine studies, for instance, Wu et al. [18] applied an inviscid flow solver for the first optimization phase of a Francis turbine runner. In the second phase, they employed a RANS model. In fact, they did not apply the high-fidelity solver actively in the optimization process. As a result, according to their reports, the optimized solution is not significantly promising, especially at off-design operating conditions.

2.3.3 SBO management techniques

Solving SBO problems needs managing some optimization aspects, such as the switching scenario between low- and high-fidelity evaluations (e.g. surrogate and high-fidelity CFD evaluations), low- and high-fidelity budgets, low-fidelity optimization or surrogate correction methods, number of design variables and their bounds in different optimization steps. Different SBO management techniques have been developed. A few of them are briefly described in this subsection.

Approximation Model Management Optimization (AMMO) [73] employs a variable fidelity technique using a trust region approach [74] and quadratic approximations. The optimizer receives the objective function and constraint values and their sensitivities from the low-fidelity evaluations. As a first-order optimization method, the response of the low-fidelity model is corrected to satisfy zero- and first-order consistency conditions with the high-fidelity model, which can guarantee the convergence beside trust region usage. To increase the convergence rate of the algorithm, the high- fidelity Hessian can be used to satisfy the second-order consistency. For complicated industrial problems, the Hessian can be replaced with approximation methods (e.g. finite difference); consequently semi-quadratic convergence can be obtained. Figure 2-12 shows the AMMO algorithm flowchart.

For instance, Alexandrov et al. [75] employed this methodology to design a 2D airfoil using RANS and Euler equations respectively for high- and low-fidelity evaluations.

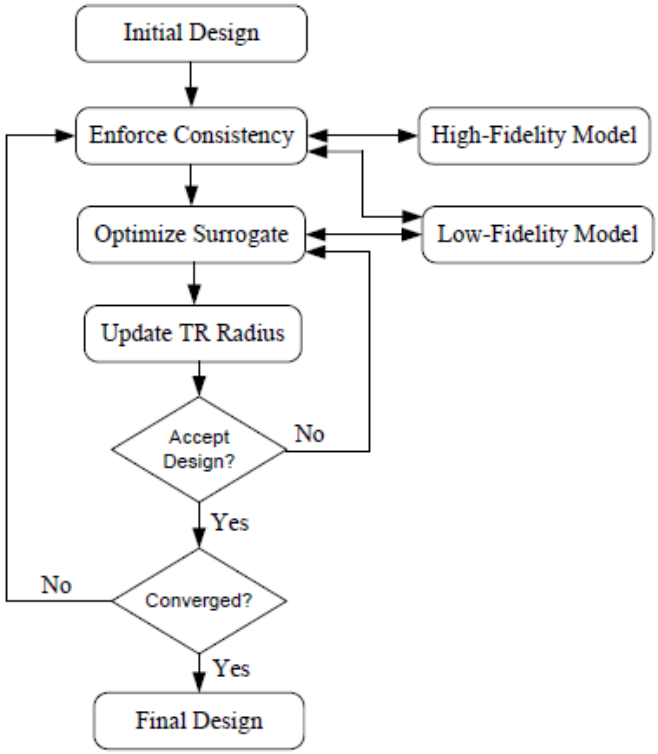


Figure 2-12 : AMMO flowchart [49]

Some SBO techniques do not require sensitivity information. For instance, Efficient Global Optimization (EGO) [65] uses a zero-order optimization strategy employing Bayesian approach along with a Kriging surrogate [76].

The convergent Trust-Region Model Management (TRMM) methodology [77] has been developed for variable-parameterization design methods. This SBO technique uses low-fidelity models and low-fidelity design spaces. Mathematical relations are defined between design vectors by a mapping method, called Space Mapping (SM) [78, 79]. To have an efficient SM optimization algorithm it is really important to have a computationally cheap but sufficiently accurate low-fidelity model. The initial SM techniques were based on a linear correction of the coarse model design space, called input SM [78]. Instead of reshaping the model domain in input SM, the model response can be corrected, called output SM [80], or the overall model properties can be changed, called implicit SM [80]. Manifold mapping (MM) [81] is a particular case of output SM, which can be expected to converge if the model response is smooth. The manifold-mapping model

alignment is illustrated in Figure 2-13 for a least squares optimization problem. In this figure, S^* constitutes a linear correction applied to the coarse model (i.e. low-fidelity model) response, R_c, to map the point $R_c(x^*)$ to $R_f(x^*)$, and to map the tangent planes of two responses at the minimizer point [61].

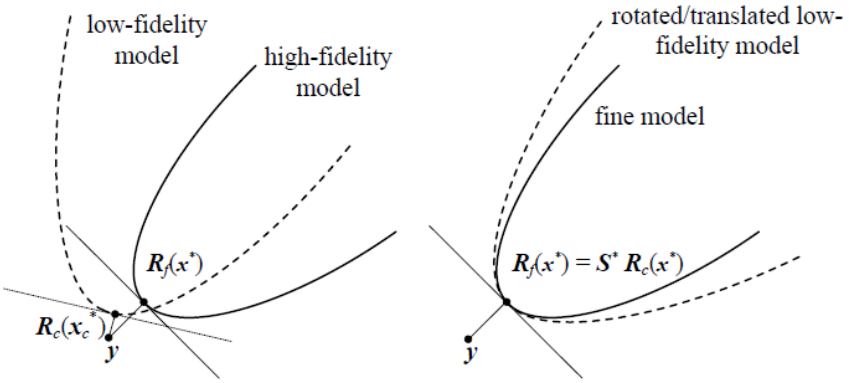


Figure 2-13 : Illustration of the manifold-mapping model alignment [61]

Another well-known SBO technique is Surrogate Management Framework (SMF) [38], which was developed for the pattern search optimization. This derivative-free mesh-based technique constitutes the MADS algorithm's ancestor, which was described in Subsection 2.2.4.

2.3.4 Infill methods

In functional surrogates local or global response approximations are built on a set of data points obtained by high-fidelity evaluations. The initial surrogate model can be created based on early evaluations of some designs chosen randomly or using smarter methods such as Latin Hypercube. Then, this model needs to be tuned step by step in order to obtain an accuracy enhancement. New high-fidelity data points called infill points are responsible for this task. The selection policy of infill points (so-called infill criteria) can determine important characteristics of the SBO such as local or global accuracy, and consequently local or global search capabilities.

Infill criteria are different in physics-based surrogate optimization, since evaluations of infill points are not supposed to improve the surrogate. In fact, physics-based surrogate approximations are not correctable, whether the approximations are in the solver physics (i.e. physical assumptions) or in the numerical processes (i.e. coarser discretization and partial convergence).

Infill methods are the key elements of multi-fidelity optimizations, since they can be the only connectors of low- and high-fidelity evaluations. Infill methods can be divided into two general

categories: exploration and exploitation methods. The exploitation methods use surrogate minima as new infill points. This method has been used in AMMO [73] and SMF [82]. The exploration methods select new infill points between the existing ones mainly in the sake of global search improvement [60].

In a complex shape optimization problem with some local minima, a combination of both infill methods is preferable. In one hand, over exploration is a waste of resources. In the other hand, exploitation in limited zones of the design space without sufficient exploration in advance can prevent finding good global solutions. Therefore, different methods have been developed to combine two infill concepts to determine a proper balance of search budget. The most popular ones are two-stage approaches [60], such as work of Jones et al. [65].

2.4 Summary

Different types of methods and models are available in the literature, which can be chosen to evaluate hydraulic turbine performance depending on the desired fidelity level and specific problem requirements. Among those methods, designers usually employ CFD evaluations in the iterative design optimization process. CFD tools have provided various choices for the flow field calculations with different levels of fidelities. For high-fidelity CFD analyses, two-equation RANS models provide a good compromise between accuracy and computational cost and time. Among them, k-ε models have been chosen for the present work. They have been widely implemented and validated in a wide range of turbine operating conditions and currently are utilized by designers in high-fidelity CFD solvers.

There are several choices of low-fidelity models as well; potential or Euler flow simulations can be employed, as well as several mathematical models. In the sake of using available tools, a potential flow solver has been considered for case studies as the low-fidelity CFD solver, which has been integrated in a parameterization code developed in Andritz Hydro Canada Inc. Potential flow analyses cannot calculate the runner efficiency improvement at targeted operating conditions, which is the main design optimization goal. However, runner designers use it very often to achieve major characteristics of a good design. The biggest challenge is to identify reliable output parameters and their error margins at specific operating conditions. Valuable designers' experience can help a lot to overcome this challenge.

A blade shape optimization problem usually has difficult constraints with a complex and non-smooth relation between variables, objectives and constraints. While gradient-based optimizers can be trapped in local minima, derivative-free optimizers have been chosen. They also have the advantage of working comfortably with CFD codes for blackbox evaluations, which may have a considerable failure percentage. Among available derivative-free optimization techniques, two of them have been selected for the present work: the MADS and EA. While both of them are relatively robust and easy to work with, they have completely different features. The MADS and EA need a large number of evaluations, which usually constitutes a common drawback of derivative-free optimization methods. The SBO implementation can alleviate it. Using a potential flow solver as a physics-based surrogate in the main optimization loop reduces the computational cost. However, as it was mentioned earlier, it cannot consider viscous effects, and consequently cannot calculate the efficiency. Thus, high-fidelity CFD analyses have to be used in a proper multi-fidelity methodology to tune the low-fidelity optimization problem and to accurately evaluate the promising designs. Such a methodology has been proposed in this work. Functional surrogate models can be employed in MADS and EA as well.

Low- and high-fidelity phases are connected to each other with a kind of infill method. In a functional surrogate optimization, infill point evaluations improve the surrogate model. However, for the work at hand, infill criteria have to be different than the ones used for functional surrogates. It is because the low-fidelity CFD solver cannot be corrected and the two phases have different objective functions. An inspiration from both types of infill methods, exploration and exploitation, can be used to accomplish the task.

CHAPTER 3 ARTICLE 1: MULTI-FIDELITY SHAPE OPTIMIZATION OF HYDRAULIC TURBINE RUNNER BLADES USING A MULTI-OBJECTIVE MESH ADAPTIVE DIRECT SEARCH ALGORITHM

S. Bahrami[1], C. Tribes[1], C. Devals[2], T. C. Vu[3] and F. Guibault[2]

[1] *Mechanical Engineering Department, École Polytechnique de Montréal, Montréal, Quebec, Canada, H3T 1J4*

[2] *Computer Engineering Department, École Polytechnique de Montréal, Montréal, Quebec, Canada, H3T 1J4*

[3] *R&D Division, Andritz Hydro Canada Inc., 6100 TransCanada highway, Point-Claire, Quebec, Canada, H9R 1B9*

*Based on the paper published in Elsevier journal of "Applied Mathematical Modelling", Vol. 40, pp. 1650-1668, 2016.

Nomenclature

B_L	Lower bound of design variables
B_U	Upper bound of design variables
C	Design characteristic
C_μ	Turbulent kinetic energy constant
$C_{\varepsilon 1}$, $C_{\varepsilon 2}$	Turbulent model constants
C^*	Targeted design characteristic
G	Sieving grid size
g	Gravity
H	Height or head
I_{Obj}	Indices used for objectives
J_{Cons}	Indices used for constraints
k	Turbulent kinetic energy
k_L	Relaxation factor of characteristic limit correction
k_{OP}	Relaxation factor of operating condition correction
k_s	Shrinkage factor
k_T	Relaxation factor of design characteristic correction
$\vec{n}$	Wall normal vector
OP	Operating point
$\widetilde{OP}$	Operating point of minimum characteristic
P	Pressure
R	Cluster radius
R^N	N-dimensional Euclidean space

$\vec{r}$	Radial coordinate vector
S_{in}	Inlet swirl
U	Characteristic limit
U_i, U_j, U_k	Cartesian mean velocity vectors
$\vec{V}$	Velocity vector
$\vec{W}$	Relative velocity vector
X	State variable
x	Cartesian position
Y	A set of geometric design variables
Y^*	A set of geometries
y	Independent design variable
Z	Mapped geometry
Z^*	A set of mapped geometries
z	Vertical coordinate in cylindrical system
Greek symbols	
ρ	Density
ε	Turbulent dissipation rate
Ω	Angular velocity
$\emptyset$	Potential function
θ	Angle
μ	viscosity
μ_t	Turbulent viscosity
σ_ε , σ_k	Turbulent model constants
Subscripts	
B	Selected band of feasible solutions
bc	Boundary condition
C	Candidate
Cons	Constraint
c	Cluster
F	Feasible
i,j,k,l,m,q,t	Counting indices
N	Number of design variables
Obj	Objective
OP	Operating condition
P	Pareto front
S	Sieved
T	Number of all evaluated geometries
Superscripts	

h	High-fidelity
l	Low-fidelity

3.1 Abstract

A robust multi-fidelity design optimization methodology has been developed to integrate advantages of high- and low-fidelity analyses, aiming to help designers reach more efficient turbine runners within reasonable computational time and cost. An inexpensive low-fidelity inviscid flow solver handles most of the computational burden by providing data to the optimizer by evaluating objective functions and constraint values in the low-fidelity phase. An open-source derivative-free optimizer, NOMAD, explores the search space, using the multi-objective mesh adaptive direct search optimization algorithm. A versatile filtering algorithm is in charge of connecting low- and high-fidelity phases by selecting among all feasible solutions a few promising solutions which are transferred to the high-fidelity phase. In the high-fidelity phase, a viscous flow solver is used outside the optimization loop to accurately evaluate filtered candidates. High-fidelity analyses results are used to recalibrate the low-fidelity optimization problem. The developed methodology has demonstrated its ability to efficiently redesign a Francis turbine blade for new operating conditions.

Keywords: multi-fidelity, multi-objective optimization, computational fluid dynamic (CFD), hydraulic turbine blade, mesh adaptive direct search algorithm.

3.2 Introduction

From an energy production perspective, the moving component of a hydraulic turbine, the runner, plays a key role in its operation. Designing a runner currently relies extensively on the designer's intuition and experience. Although runner designers employ CFD tools to evaluate their designs, there is a strong need to integrate more tightly CFD analyses to obtain more automatic and efficient design processes. A full range of CFD methods has been utilized in the optimization of hydraulic turbine runner blades; from low-fidelity inviscid models (e.g. using potential flow [17]) to high-fidelity viscous models (e.g. using a turbulent RANS solver [20]). However none of these methods can, by itself, entirely fulfil industrial design needs. On one hand, low-fidelity CFD simulations are not accurate enough in their prediction of flow behavior, mainly due to shortcomings in the physics. On the other hand, high-fidelity CFD analyses cannot be used in the main optimization

loop, since they are too expensive and slow for iterative industrial blade design processes. Surrogate-based optimization approaches have been employed, whereby computationally inexpensive models are used in lieu of high-fidelity models. These approaches may be divided into functional and physics-based surrogates. Although functional surrogates have been used for blade shape optimizations (e.g. radial basis functions [83], artificial neural network [84], and Kriging [85]), they require a large number of high-fidelity evaluations to update the surrogate model, and to ensure that they yield reasonably accurate results. For instance, Georgopoulou et al. [63] applied a functional surrogate model based on radial basis functions, to optimize runner blades of hydraulic turbines using an evolutionary algorithm in a hierarchical scheme. They obtained design candidates for one test case after 3500 high-fidelity evaluations. Some researchers combined several surrogate models to reduce the number of evaluations required to obtain an accurate surrogate. In this technique, the same evaluation points are used to produce multiple optima dedicated to multiple surrogate models. This combination can eliminate the risk of wrong surrogate model selection and provide the robustness to choose the most proper model at each optimization level. For instance, recently, Badhurshah and Samad [67] incorporated response surface, radial basis function, and Kriging to optimize a bidirectional impulse turbine blade. They reported an average number of 600 iterations and an average time of 6 hours for a single simulation. Vesting and Bensow [68] also described a similar time-consuming process in a multi-surrogate optimization of a marine propeller.

In physics-based surrogates, high-fidelity evaluations are replaced by low-fidelity analyses using one of the following techniques: simplifying governing equations (e.g. Jameson and Reuther [86]), using coarser grids (e.g. Leary et al. [72]) and relaxing convergence criteria (e.g. Forrester et al. [71]). A combination of high- and low-fidelity models and techniques can also be used in a multi-fidelity framework. In such a framework, low-fidelity models are corrected by accurate high-fidelity information through correction techniques such as response correction [73] and space mapping [87]. Low-fidelity models are typically more computationally expensive than functional surrogates and are also problem dependent; but they have the big advantage of requiring much fewer high-fidelity evaluations to obtain a given level of accuracy [49]. In some cases, by having a proper surrogate, a very few high-fidelity evaluations are needed per algorithm iteration (e.g. Booker et al. [38], Robinson et al. [88] and Alexandrov et al. [89]). Several techniques have been developed to manage multi-fidelity optimization. Alexandrov and co-workers [75] introduced approximation model management optimization (AMMO) which exploits the trust region

methodology [74] to guarantee the convergence to the optimal high-fidelity solution. This technique needs objective and constraint values and their sensitivities to correct the response of the low-fidelity model to satisfy zero- and first-order consistency conditions. They employed this methodology to design a 2D airfoil using RANS and Euler equations respectively for high- and low-fidelity evaluations.

Leifsson and Koziel [53] took advantage of the shape-preserving response prediction (SPRP) method, introduced initially in the microwave engineering field [90], in a multi-fidelity airfoil design problem. They employed this technique to optimize the shape of a 2D blade controlled by three design variables. The low-fidelity model was corrected by aligning its corresponding airfoil surface pressure distribution with that of the high-fidelity model using the SPRP technique. Applying this design method to real industrial cases with significantly higher number of design variables is quite complex, and needs more investigations.

Jameson developed several multi-fidelity methods using control theory with adjoint methods to derive the gradient of a cost function with respect to the shape. For instance, Jameson and co-workers used continuous adjoint methods in multi-fidelity airfoil designs, which applied two low-fidelity models; potential flow [69], and compressible Euler flow [86]. Using gradient-based optimization methods (e.g. adjoint method) is possible when the high-fidelity model sensitivity is available and gradients can be computed at a reasonable cost.

In the hydraulic turbine field, Wu et al. [91] applied a multi-fidelity CFD-based optimization approach to a Francis turbine runner design. They applied inviscid quasi-3D and 3D Euler codes in early stages of runner optimization in an iterative process. In the final stage, a commercial Navier-Stokes code was employed to accurately evaluate only the final optimized runner. Although they used a multi-fidelity optimization, they did not apply the high-fidelity solver directly in the optimization process.

A key feature of all the aforementioned multi-fidelity methods is the fact that the high- and low fidelity phases are connected through the use of the same objective functions in both phases. This connection is required in order for their proposed correction techniques to be usable. However, quite often in industrial problems, the high-fidelity objectives are not assessable with low-fidelity evaluations. The main contribution of this paper is proposing a new multi-fidelity optimization methodology, which addresses those types of problems where the high- and low-fidelity objectives

are different. This methodology employs an optimization correction that tunes the low-fidelity optimization problem with high-fidelity data. Also, a unique filtering method connects the two phases by selecting a given number of promising candidate solutions and transferring them to the high-fidelity phase. Another contribution is splitting and managing the evaluation budget and resources between the two phases, for a better consideration of two important industrial design limitations, time and cost. Different budget balances may be chosen in different fields based on various criteria, such as the level of accuracy and cost associated with each phase. It gives different designers the necessary flexibility to adequately balance resource consumption according to the specific characteristics of their applications. Although splitting computational burden has been used in some previous works using hierarchical algorithms (e.g. [64]), the high-level evaluations have been mostly applied to correct surrogate models in the low-levels. However, in the methodology proposed in the present paper, a physics-based surrogate is employed, which cannot be corrected. This feature distinguishes the present approach from previously proposed hierarchical approaches.

The developed multi-fidelity optimization methodology and formulation are described in Section 2. For demonstration, the methodology is applied in a hydraulic turbine optimization problem. In the next section, numerical methods are explained, which are employed in the test case optimization. This field has been chosen since there is no truly cost effective method applicable to hydraulic turbine design. It is mainly because hydraulic turbine designers widely use inviscid flow solvers to handle most of design process evaluations, which are unable to evaluate the viscous energy losses. In Section 4, the test case details are explained, including the problem formulation, parameterization, and discussion on the results. We previously presented briefly a similar case study using a single operating condition [92]. However, in the present paper, full details of the multi-fidelity methodology and its components are formulized and explained for the first time, and two operating conditions are considered in the hydraulic runner blade optimization. Also, target correction is demonstrated using two different scenarios.

3.3 Design optimization methodology

The developed multi-fidelity optimization methodology consists of three components. Optimization is performed only in the low-fidelity phase (part A in Figure 3-1) using low-fidelity analysis results. A filtering algorithm (part B in Figure 3-1) identifies promising candidates among

the optimization solutions. The high-fidelity phase (part C in Figure 3-1) serves to identify the best design candidate using high-fidelity analyses. Based on some convergence criteria, this design may be selected as the final design or be transferred to the low-fidelity phase as the new initial design. Another purpose of the high-fidelity phase is correction of the low-fidelity optimization problem. The aforementioned components of the methodology are defined and their roles are explained in the following sections.

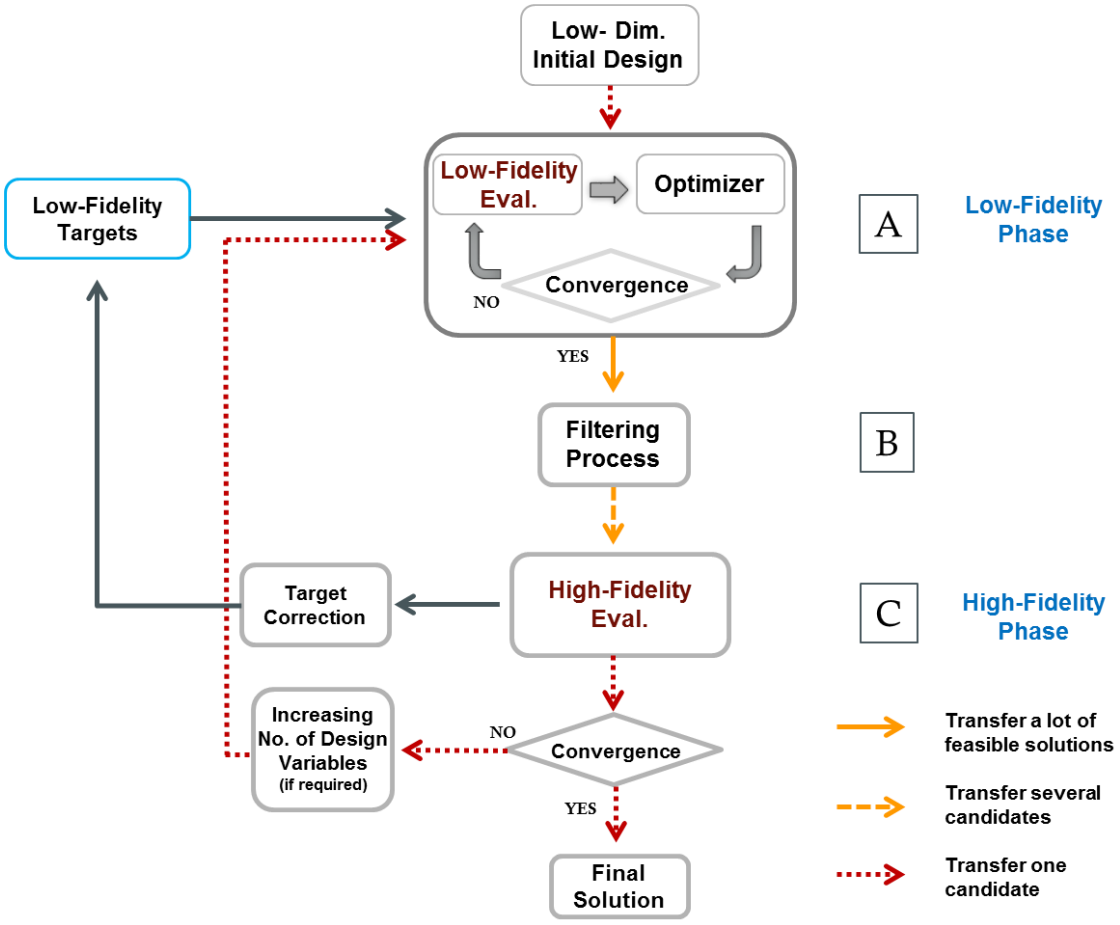


Figure 3-1 : Multi-fidelity design optimization algorithm

3.3.1 High-fidelity phase

Let us consider a design problem for which the independent design variables are only geometric variables defining the shape of mechanical equipment. For a given geometry defined as a vector (noted Y), the state variables of a high fidelity analysis (noted X^h) are obtained by solving some governing equations for some given boundary conditions (noted X_{bc}^h), which can be written in a residual form as follows:

$$R^h\left(X^h, X_{bc}^h, Y\right) = 0 \tag{3-1}$$

The state variables are outputs of the analysis and they are dependent variables of the boundary conditions and the geometry, which is written in a functional form as follows:

$$X^h = X^h\left(Y, X_{bc}^h\right) \tag{3-2}$$

The main characteristics of the design can be determined from the state variables and expressed as:

$$C^h = C^h\left(Y, X_{bc}^h\right) \tag{3-3}$$

The main characteristics serve as high-level performance measures of a design. For example, the drag, lift and moment coefficient are important measures of a wing performance. The design problem considers some operating points, each given as a set of single-valued parameters. All characteristics can be evaluated for a given operating point, which is written in a functional form:

$$C_i^h = C_i^h\left(Y, OP_i^h\right) \tag{3-4}$$

To obtain the desired characteristics at the proper operating point, OP_i^h the boundary conditions X_{bc}^h must be adjusted adequately while running analyses iteratively. For example, to obtain the flow characteristics around a fuselage and its wings that can support the whole aircraft weight, the angle of attack must be adjusted to obtain the proper lift. Hence, the angle of attack will not be the same whether a high-fidelity Navier-Stokes flow solver or an Euler flow solver is used, because the flow characteristics obtained by those two solvers are different. The same situation happens for a Francis turbine, for which the operating condition dictates an output power by the machine that depends on the flow characteristics and the guide vane outflow angle.

Let us suppose that the shape design problem can be stated as an optimization problem as follows:

$$\min_{Y \in \Omega}\left[C_i^h\left(Y, OP_i^h\right)\right]_{i \in I_{Obj}^h} \tag{3-5}$$

where Ω is the set of feasible solutions defined as:

$$\Omega = \left\{Y \in R^N \mid C_j^h\left(Y, OP_j^h\right) \leq U_j^h;\ \forall j \in J_{Cons}^h\right\} \tag{3-6}$$

The limits U_j^h for some of the characteristics are also part of the design problem definition like the operation points. Many engineering design problems using complex analyses can be stated as in Eq. 3-5.

In most cases, using high-fidelity analyses to obtain all needed characteristics is costly and solving the corresponding optimization problem becomes prohibitively time consuming. While substitution towards using low-fidelity analysis becomes an attractive alternative, mapping low-fidelity characteristics C_i^l with high-fidelity characteristics C_i^h poses some challenges. The proposed methodology applies to a situation where some of the high-fidelity characteristics C_i^h used for high-fidelity objective/constraint evaluations are not output by the low-fidelity analysis, because of the lack of physics involved; but other important characteristics are sufficiently well predicted. For instance, Euler flow model cannot predict viscous drag on a surface, but the pressure distribution can be reasonably well predicted in some situations, and can be used to define a good low-fidelity surface loading. Thus, some targets, C_i^* , can be set by experienced designers for a subset of the characteristics assessable by low-fidelity analyses. Experienced designers are accustomed to informally map good targets C_i^* for a subset of characteristics to overall good designs. In practice, designers cannot always find a design that matches all the targets. Nevertheless, designs that are sufficiently close to the targets can be considered as good candidates. This optimization methodology introduces automation for the low fidelity optimization problem and its correction.

3.3.2 Low-fidelity phase

The state variables X^l are outputs of the low-fidelity analysis. They can be obtained by solving the low-fidelity governing equations for some given boundary conditions (noted X_{bc}^l) and the geometry. A set of boundary conditions is associated with an operating point, OP_i^l. Therefore, similar to what was done in Section 2.1, the design characteristics can be written as dependent variables of the geometry and the operating points:

$$C_i^l = C_i^l\left(Y, OP_i^l\right) \tag{3-7}$$

Let us denote $Y^* = \{Y_1, \dots, Y_t, \dots, Y_T\}$ the set of all geometries that have been evaluated while solving the following optimization problem:

$$\min_{Y\in\Omega}\left[f_i\left(Y,OP_i^l\right)\right]_{i\in I_{Obj}^l} \tag{3-8}$$

where Ω is the set of feasible solutions defined as:

$$\Omega=\left\{Y\in R^N|\ B_L\le Y\le B_U \text{ and } C_j^l\left(Y,OP_j^l\right)\le U_j^l;\ \forall j\in J_{Cons}^l\right\} \tag{3-9}$$

and:

$$f_i\left(Y,OP_i^l\right)=\left\|C_i^*-C_i^l\left(Y,OP_i^l\right)\right\| \tag{3-10}$$

3.3.3 Filtering process

The optimal solutions obtained in the optimization are not necessarily the dominant ones from the high-fidelity perspective, since low-fidelity evaluations have been used during the optimization. Therefore, a special filtering process is required to select a certain number of feasible potential candidates (noted Y_C^*) for which $C_i^l\left(Y,OP_i^l\right)$ is not too far from C_i^*. For a better exploration of low-fidelity optimization solutions, those potential candidates need to be geometrically different enough. These critical steps are described in Section 3.3. Promising candidates are evaluated with high-fidelity analyses to obtain $C_i^h\left(Y,OP_i^h\right)$ for all operating points.

3.3.4 Modifications of low-fidelity optimization problem

The low-fidelity optimization problem given in Eqs. 3-8 and 3-9 can be modified if the budget allotted to the design process allows repeating the low-fidelity phase using a new initial geometry and corrected optimization features. Specifically, if none of the filtered design candidates is good enough or if further improvements are expected, the low fidelity optimization problem can be modified by: adjusting the low-fidelity operating points OP^l, changing the characteristic targets C^*, and/or modifying the limits on characteristics U^l. Furthermore, the design space may be refined.

Linear corrections are used in this investigation based on the gaps between the results obtained from high- and low-fidelity analyses. The corrections can be effective only if the low-fidelity optimization problem is adequately representative of the high-fidelity optimization problem (i.e.

Eqs. 3-5 and 3-6). However, it is not guaranteed even when C_i^l and C_i^h are relatively similar. Therefore, we rely on the best candidate improvements after applying the corrections, in order to continue iterating.

4.3.4.1 Changing operating points

The targeted operating point is fixed. It is considered as the operating point used by the low-fidelity analysis OP_i^l in the first step of optimization. For the next optimization step, OP_i^l can be corrected by identifying the minimum value of the high-fidelity characteristic C_i^h with respect to OP. For instance, we consider minimizing C_i^h at a single-valued OP_i^h. However, for a given geometry, high-fidelity analysis results indicate that the minimum is at $\widetilde{OP}_i$ (see Figure 3-2). We want to find good candidates for which the minimum value of C_i^h will be shifted from $\widetilde{OP}_i$ to OP_i^h. A first-order correction of the low-fidelity operating point may be considered:

$$OP_i^l \leftarrow OP_i^l + k_{OP}\left(OP_i^h - \widetilde{OP}_i\right) \tag{3-11}$$

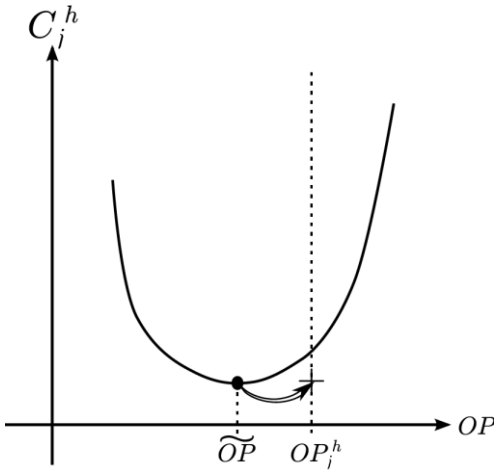


Figure 3-2 : Illustration of operating point correction

where the relaxation factor k_{OP} is taken between 0 and 1. This type of correction is based on the idea that the operating point can be shifted by approximately the same amount for low and high fidelity analyses.

4.3.4.2 Changing characteristic targets

Let us consider a best design candidate such that $C_i^l \approx C_i^*$, but there is a gap between C_i^l and C_i^h. Then, a correction on C_i^* can be determined by considering the aforementioned gap:

$$C_i^* \leftarrow C_i^* + k_T\left(C_i^l - C_i^h\right) \tag{3-12}$$

where the relaxation factor k_T is taken between 0 and 1. Figure 3-3 illustrates the case where a characteristic C_i is obtained for some position r. In this case, the low fidelity analysis has obtained values of C_i^l that are systematically higher than C_i^h. Therefore, the target curve has to be shifted up to correct this trend and obtain higher values of C_i^h. A smoothing of the corrections may be required (e.g. when a linear target curve is preferred).

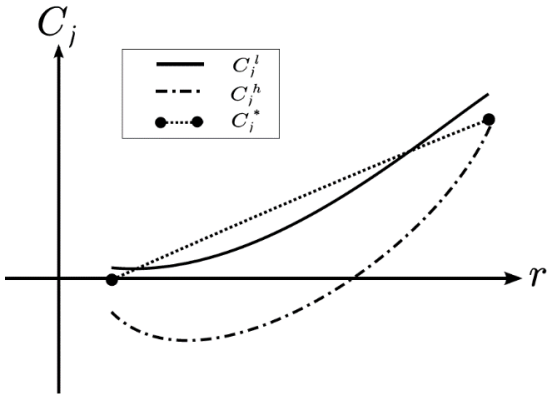


Figure 3-3 : Illustration of target correction

4.3.4.3 Changing characteristic limits

If for the best design candidate, there is $C_j^l \leq U_j^l$ (as a result of low-fidelity optimization), but high-fidelity evaluations indicate $C_j^h \geq U_j^h$, a correction on U_j^l can be determined:

$$U_j^l \leftarrow U_j^l + k_L\left(\left(U_j^l - C_j^l\right) + \left(C_j^h - U_j^h\right)\right) \tag{3-13}$$

where the relaxation factor k_L is taken between 0 and 1.

4.3.4.4 Design space modifications

The design space can be modified as well by adjusting the bound constraints of design variables (B_U and B_L in Eq. 3-9). Sometimes the characteristics C_i^l of the best design candidate are too far from C_i^* or a few feasible solutions have been obtained during the low-fidelity phase. Therefore, the dimension of the design variable space can be increased as well, in order to provide more flexibility to the optimizer. When changing the dimension of the problem, the geometry of the current best design candidate should be mapped from the old design space into the new design

space. When possible, this mapping allows identifying the initial values of the design variables to restart the low-fidelity optimization. This is possible only if the parameterization software supports such transformation. The bounds on design variables may be changed mostly when several design variables for the best design candidate lie on their bounds.

3.4 Numerical methods

A hydraulic turbine runner optimization problem has been chosen to demonstrate the ability of the proposed methodology to handle a real industrial problem. A good hydraulic turbine blade has to meet some criteria from different disciplines such as hydrodynamic, mechanical stress, and fluid-structure interaction. In this study, only the fluid dynamic perspective is considered. This section introduces the numerical methods employed in the proposed methodology for solving this problem.

3.4.1 Potential flow analysis

In this project, a potential flow solver handles the low-fidelity computational burden. However, other low-fidelity analyses may be used based on different design cases. Since the flow is assumed to be inviscid, incompressible and irrotational, the absolute velocity vector can be obtained by the gradient of the potential:

$$\vec{V} = \nabla \emptyset \tag{3-14}$$

From conservation of mass, Laplace's equation is obtained:

$$\nabla^2 \emptyset = 0 \tag{3-15}$$

The pressure, P , can be calculated from:

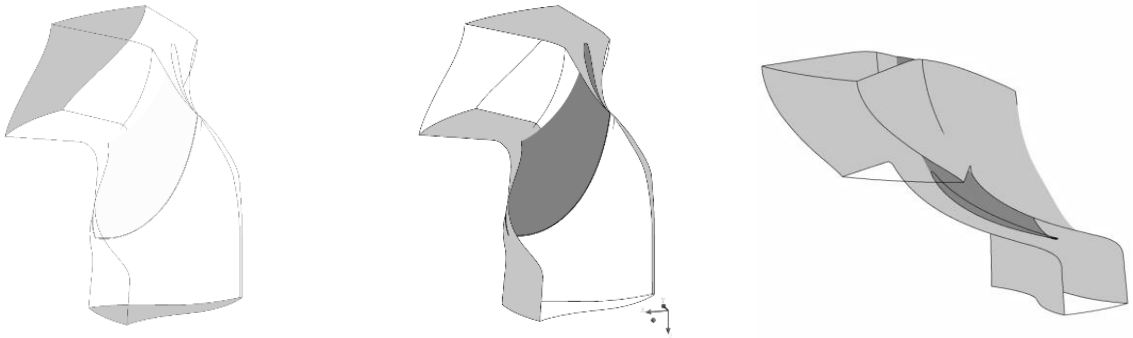
$$P = \rho g \left[\Delta H - \frac{V^2}{2g} + \Omega(S_{in} - rV_\theta) + z \right] \tag{3-16}$$

The inlet swirl S_{in} is equal to $\frac{\partial \emptyset}{\partial \theta}$ [17]. Potential distributions are specified by Dirichlet conditions on the inlet and outlet surfaces. Inlet and outlet surfaces have been illustrated in Fig. 3a respectively at the top and the bottom of the blade. On solid walls (Fig. 3-4b), the boundary conditions are given as:

$$\overrightarrow{W}.\vec{n} = 0 \quad (3\text{-}17)$$

$$\overrightarrow{W} = \vec{V} - \vec{\Omega} \times \vec{r} \quad (3\text{-}18)$$

One fluid flow channel containing one blade is selected, and periodic boundary conditions are imposed to two sides of the domain to obtain the complete runner domain (Figure 3-4c). This boundary condition can be applied by requiring constant potential jumps between corresponding points on the upstream and downstream pairs of periodic surfaces, which are set respectively by inlet and outlet potential distributions [17]. A finite element grid is employed using cubical elements with a node at each intersection of three grid lines. More details of the finite element algorithm is available in Holmes and McNabb's paper [17].



(a) (b) (c)

Figure 3-4 : Computational domain boundaries: (a) inlet and outlet, (b) walls, (c) periodicity

3.4.2 Mesh Adaptive Direct Search (MADS) Optimization Method

In this project, a derivative-free optimization algorithm has been selected to conduct a black-box optimization, which involves non-convex and discontinuous functions and a noisy potential flow solver. NOMAD (Non-smooth Optimization by Mesh Adaptive Direct Search) [93] is an open source [94] C++ implementation of the Mesh Adaptive Direct Search algorithm [95, 96]. NOMAD has demonstrated its ability to handle different optimization problems in several fields, such as thermo chemical problems [97] and optical metamaterial problems [98]. In directional direct search (e.g. MADS) methods a set of directions with suitable features are used to generate a finite set of points at which the objective function is evaluated [99]. The MADS algorithm iterates to improve the current best solution on a mesh of varying refinements, which is a discretization of the design

space by NOMAD. When the iteration results in no improvement, the next iteration will be initiated on a finer NOMAD mesh. Each iteration is composed of two steps: the Search and the Poll. The Poll generates trial mesh points in the vicinity of the best current solution, and this algorithmic feature provides the basis for convergence analysis of the overall optimization process [48].

The Search can return any point on the underlying mesh in order to improve the current best solution. The default Search step uses a quadratic model of all functions by using available evaluations and conducting an optimization on this model. In addition, a variable neighborhood search (VNS) is employed to escape from local minima. The VNS search strategy is described in [50].

The main convergence criterion of MADS algorithm is met when the mesh size becomes small enough. However, in practice we also set the maximum number of evaluations to stop the optimization with respect to available computational resources. For this optimization problem with two objective functions, the BIMADS algorithm [51] has been used. It solves series of single-objective reformulations of the bi-objective problem using the MADS algorithm [52]. An approximation of the Pareto front, or a set of points that are dominant, is constructed with the evaluations performed during those MADS runs [48].

3.4.3 Filtering method

A novel filtering method for physics-based surrogate optimization has been developed, which imposes no significant computational burden. The characteristics of good design candidates are evaluated through the high-fidelity phase. Some feasible solutions (i.e. new blade geometries that satisfy the constraints) obtained from the optimization loop of the low-fidelity phase can be potentially considered as good candidates and be evaluated in the high-fidelity phase. However, just a few of them can be investigated by costly high-fidelity evaluations. The developed filtering method consists of several filtering steps and one mapping step, which are described in the following subsections.

4.4.3.1 Pareto front determination

In the first step, feasible solutions, Y_F^*, are filtered from all solutions of the optimization formulated in Eqs. 3-8 and 3-9 of Section 2.

$$Y_F^* = \{Y_1, \dots, Y_f, \dots, Y_F\} \quad (3\text{-}19)$$

$$Y_f = \{y_{f,1}, \dots, y_{f,n}, \dots, y_{f,N}\} \quad (3\text{-}20)$$

The corresponding objective functions to minimize are:

$$f(Y) = [f_1(Y), \dots, f_i(Y), \dots, f_I(Y)] \quad (3\text{-}21)$$

In a multi-objective optimization, instead of one optimal solution, there is a set of optimal points called the Pareto front. In the first part of the filtering process, Pareto solutions are identified among feasible solutions based on a dominance criterion [100]. A solution, Y_P , is called a Pareto optimal solution if there is no $Y \in Y_F^*$ (except Y_P) satisfying

$$f_i(Y) \leq f_i(Y_P) \ ; \ \text{for all} \ \ i = 1, 2, \dots, I \quad (3\text{-}22)$$

Preliminary investigations have shown that other solutions that are not too far from the Pareto front can also have high quality, which is the final goal of the design process. Therefore, a band of feasible solutions is selected (i.e. $Y_B^* \subseteq Y_F^*$), based on the distance from the Pareto front. This band is limited to Pareto front on one side, and an offset of Pareto on the other side. The offset is reached by shifting the Pareto curve obtained using a proper regression. The shifting vector determines the width of the band, and can be adjusted based on the distribution and concentration of feasible solutions in the objective space, in order to obtain more or less solutions in Y_B^*.

4.4.3.2 Mapping and sieving

In the next step, filtering is performed based on the difference (distance) between solutions in the design variable space. To allow that, it is necessary to have a proper working space to easily recognize the amount of similarities/dissimilarities of solutions in a standard design space.
The set of solutions, Y_B^*, is mapped into $Z_B^* = \{Z_1, \dots, Z_b, \dots, Z_B\}$ using the following equation:

$$Z_{b,n} = \frac{G \times \left(y_{b,n} - \min\limits_{b=1,\dots,B} y_{b,n}\right)}{\max\limits_{b=1,\dots,B} y_{b,n} - \min\limits_{b=1,\dots,B} y_{b,n}} \quad (3\text{-}23)$$

Where “G” is the given number of divisions for each dimension. A similar mapping was applied in the research of Postaire et al. [101] to prepare a set of observations for binary morphological

transformations. This mapping allows dividing the portion of the design space into hypercubes of unit length sides. In the next step, the sieving process selects the dominant solution of each mutual-solution hypercube (containing more than one solution), which is closest to the hypercube center. By this methodology, the most promising solutions that are globally dissimilar enough are selected. After the sieving process, $Z_S^* \subseteq Z_B^*$ is obtained. Figure 3-5 shows an example of applying the sieving process to a two-dimensional design space with two objective functions.

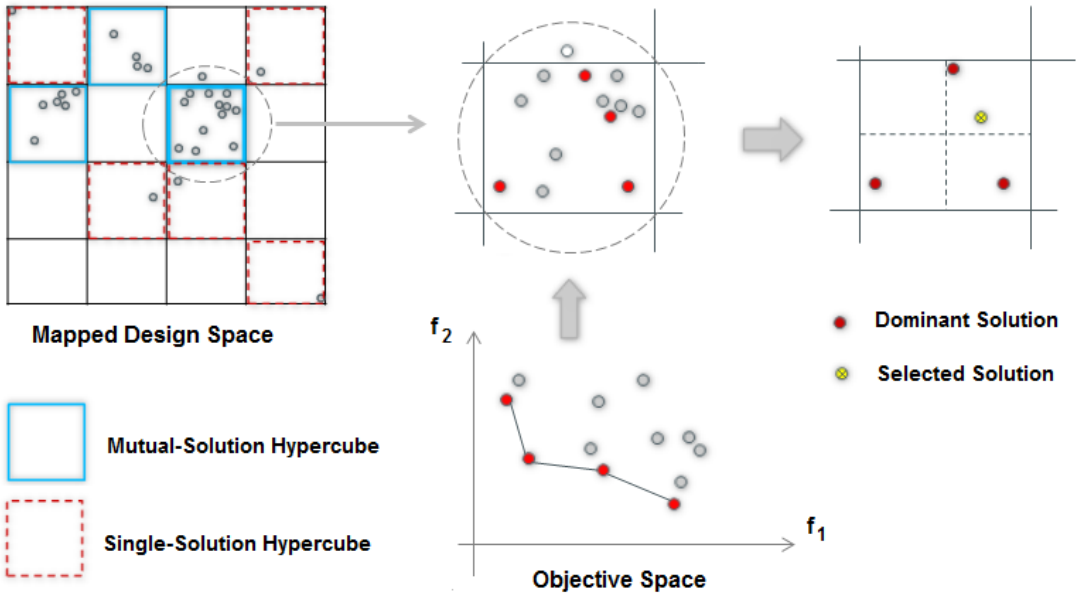


Figure 3-5 : Sieving process; selecting one candidate from each hypercube

4.4.3.3 Cluster formation

After the sieving process, the number of candidates can still be high. Based on the concept of core territories developed by Dominique et al. [102], a simple inexpensive clustering method has been developed to select a desired number of candidates, "C" (i.e. $Z_C^* \subseteq Z_S^*$, where $Z_C^* = \{Z_1, \dots, Z_c, \dots, Z_C\}$). Every candidate belongs to a cluster territory delimited by a hypersphere. All hyperspheres have the same radius. The centers (i.e. cores) of hyperspheres are selected as the final candidates. The proposed method needs an initial core and an initial radius to start. New cores are chosen by using a loop over the entire set of candidates in Z_S^*. Each candidate located outside the previous cluster territories is selected as a new core. The radius is modified in an iterative process to come up, in the end, with the specified number of candidates. The general structure of the clustering algorithm is as follow:

Select Z_1 in Z_S^* (the first cluster center)

Initialize R (cluster radius, calculated from Z_1 distances from design space borders)

While $|Z_C^*| \neq C$

Begin

Initialize Z_C^* by Z_1

For each $Z_s \in Z_S^*$

If Z_s is outside Z_C^* territories

put Z_s in Z_C^*

End

End

Update R: $R = R \times k_s \left(\frac{|Z_C^*|}{C}\right)$ (k_s: shrinkage factor)

End

3.4.4 Navier-Stokes analysis

High-fidelity flow characteristic results can be obtained from different types of investigations such as experimental testing or CFD evaluations. In this project, the commercial Navier-Stokes code ANSYS-CFX has been used for the flow field simulations. The viscous flow analysis has been performed only for a single passage of the runner flow domain since rotational periodic conditions are valid. It has been discretized using approximately a 200,000 structured grid, hexahedral control volumes. The steady Reynolds time-averaged continuity and Navier-Stokes equations in a rotating reference frame for incompressible flow can be written as:

$$\frac{\partial \rho U_i}{\partial x_i} = 0 \tag{3-24}$$

$$\frac{\partial \rho U_i U_j}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j}\left((\mu + \mu_t)\frac{\partial U_i}{\partial x_j}\right) + \rho g_i - \rho \varepsilon_{ijk}\varepsilon_{klm}\Omega_j\Omega_l x_m - 2\rho\varepsilon_{ijk}\Omega_j U_k \tag{3-25}$$

Where x_i is the Cartesian position vector, U_i is the mean relative Cartesian velocity vector, and Ω_j is the angular rotation vector. The rotating coordinate frame introduces two additional terms; $-\varepsilon_{ijk}\varepsilon_{klm}\Omega_j\Omega_l x_m$ which is the centripetal term, and $-2\varepsilon_{ijk}\Omega_j U_k$ which is the Coriolis term. ε_{ijk}

is Levi-Cevita's permutation function [103].

The standard two-equation k-ε RANS turbulence model is used because of its reasonable precision and robustness to solve the turbulent flow in the runner. In this turbulence model, turbulent viscosity, μ_t , is correlated with turbulent kinetic energy, k , and its dissipation rate, ε:

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{3-26}$$

where C_μ is a constant turbulent quantity. k and ε are calculated from partial differential transport equations [104]:

$$\frac{\partial \rho U_j k}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right) + P_k - \rho\varepsilon \tag{3-27}$$

$$\frac{\partial \rho U_j \varepsilon}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right) + \frac{\varepsilon}{k}(C_{\varepsilon 1} P_k - C_{\varepsilon 2}\rho\varepsilon) \tag{3-28}$$

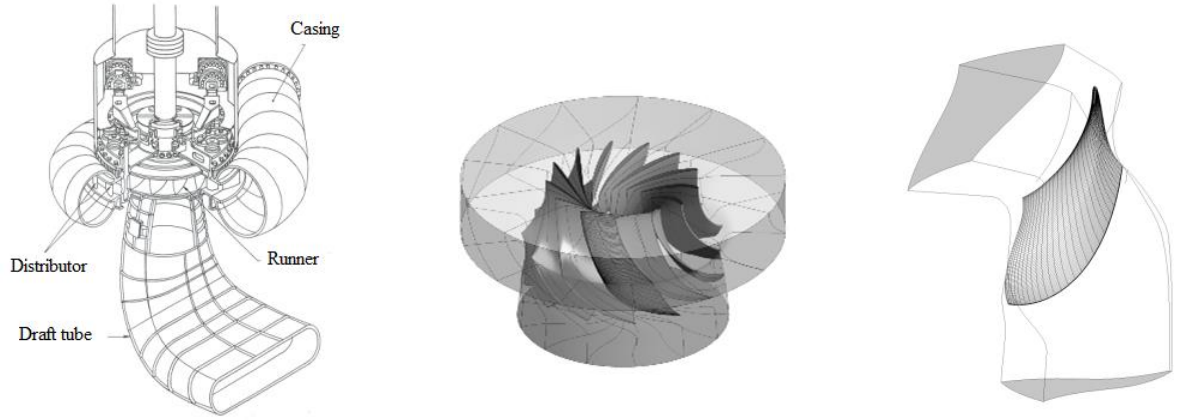
where $C_{\varepsilon 1}$, $C_{\varepsilon 2}$, σ_k and σ_ε are constants and P_k is a production term which depends on the viscous stress tensor and velocity gradients.

The high-resolution scheme and first-order upwind scheme have respectively been used to discretize the momentum and turbulent equations. Various operating conditions are simulated by using a wide range of wicket gate opening angles, from 8 to 34 degrees. Different wicket gate opening angles lead to different velocity profiles as runner inlet boundary conditions. All boundary locations are the same as those of potential flow evaluations, which was shown in Fig. 3-4. A zero total pressure is applied as the outlet boundary condition. Blade, crown and band boundaries are defined as non-slip walls. Rotational periodic boundary conditions are imposed to the two remaining surfaces. Details of the methodology, integrated tools and validations can be found in reports by Gauthier [105] and investigations by Vu et al. [106, 107]. In order to determine the efficiency of the whole turbine, it is also necessary to consider losses of other components of the turbine (e.g. casing, distributor and draft tube). Losses of other components, computed previously in a wide range of operating conditions, are added to the actual runner loss for the calculation of the turbine efficiency. Post-processing of CFD evaluations indicates which candidate has the best performance in the targeted operating point. It can be selected as the final optimized design,

according to certain convergence criteria (e.g. obtaining negligible efficiency improvement at the desired best efficiency point, BEP, position in the given tolerance) or computational budget limits.

3.5 Test case

A medium-head Francis turbine runner was chosen as the test case to evaluate the developed methodology. Figure 3-6 (a) shows a schematic of Francis turbine components. A goal was to redesign and optimize an existing efficient runner for new operating conditions. The goal was to design a new runner such that it provides its BEP at 12% higher power coefficient, while keeping the same speed coefficient condition. In addition, it was aimed to prevent cavitation at a given full-load condition (29% higher power coefficient and 5.1% higher speed coefficient than the targeted BEP).



(a) (b) (c)

Figure 3-6 : (a) Francis turbine components. (b) Runner flow domain. (c) Single-blade computational domain

This turbine contains 15 runner blades. Figure 3-6 (b) shows the aggregated fluid flow domain considered in the computational analyses, including all passages around the blades. One of those flow passages is shown in Figure 3-6 (c) containing a single blade. Inlet and outlet boundaries are illustrated on the top and bottom of the blade passage respectively.

3.5.1 Optimization problem formulation

The general optimization formulation was presented in Section 2. Here, a bi-objective hydraulic blade optimization formulation is proposed, which can be easily fitted to different turbo machinery shape optimization problems.

$$\min [f_i(Y)] \quad ; \quad \text{i=1, 2} \tag{3-29}$$

$$\text{Subject to} \quad C_j^l \leq U_j^l \quad ; \quad \text{j=1, 2, 3, 4} \tag{3-30}$$

Velocity profile distribution at the runner outlet significantly affects the draft tube operation (specially swirl), which plays a key role in the overall turbine performance. To control the losses in the draft tube, it is necessary that the runner deliver an appropriate flow velocity profile at the draft-tube inlet [10, 108]. Two objectives associated with tangential and axial velocity target profiles at the runner outlet have been defined, which lead to an inverse optimization problem. The objective functions are defined as the distance between the velocity profiles obtained by low-fidelity analyses $V(y)$, and the target profiles V^* determined on a set of discrete radii r_k. For instance, the tangential velocity objective is written as:

$$f_{Tan}(y) = \sqrt{\sum_{k=1}^{K} \left((V_{Tan}^*)_{r_k} - \left(V_{Tan}(y)\right)_{r_k} \right)^2} \tag{3-31}$$

Table 3.1 : Problem formulation

Characteristic	Objective	Constraint	Operating point
Tangential velocity profile	$f_{Tan}(y)$	-	OP_1
Axial velocity profile	$f_{Axial}(y)$	-	OP_1
Average load slope	-	$C_{LS,Max}(y) \leq U_{LS,Max}$	OP_1
		$-C_{LS,Min}(y) \leq -U_{LS,Min}$	OP_1
Smooth load distribution	-	$C_{SLD}(y) \leq U_{SLD}$	OP_1
Cavitation	-	$-C_{Cav}(y) \leq -U_{Cav}$	OP_2

Four inequality constraints have been considered (see Table 3.1). Having a smooth blade loading from leading edge to trailing edge and from band to crown is an important design criterion to obtain a good blade hydraulic performance. It has been addressed by three constraints. Two of them have been defined to limit minimum and maximum load slopes among all sections on pressure and

suction sides (i.e. $C_{LS,Min}$, $C_{LS,Max}$). For instance, the maximum allowed average slope of blade loading is constrained by $C_{LS,Max}(y) \leq U_{LS,Max}$, where $C_{LS,Max}(y) = \max_{i=1,\dots,Q} C_{LS,i}(y)$. The average slope is a positive value calculated for each of "Q" blade sections. The third constraint controls the smoothness of the load distribution at different blade sections (i.e. C_{SLD}) by considering a special load distribution target (i.e. C^*_{SLD}) for each of those sections (e.g. section 7 in Figure 3-7). Each section load target is dynamically defined as a linear fitting of the current section loading. This constraint forces the dispersal distance of two load curves to be less than a given maximum distance value, $C_{SLD}(y) \leq U_{SLD}$, where:

$$C_{SLD}(y) = \sqrt{\sum_{k=1}^{K}\sum_{q=1}^{Q}\left(\left(C_{SLD,q}(y)\right)_{r_k} - \left(C^*_{SLD,q}(y)\right)_{r_k}\right)^2} \qquad (3\text{-}32)$$

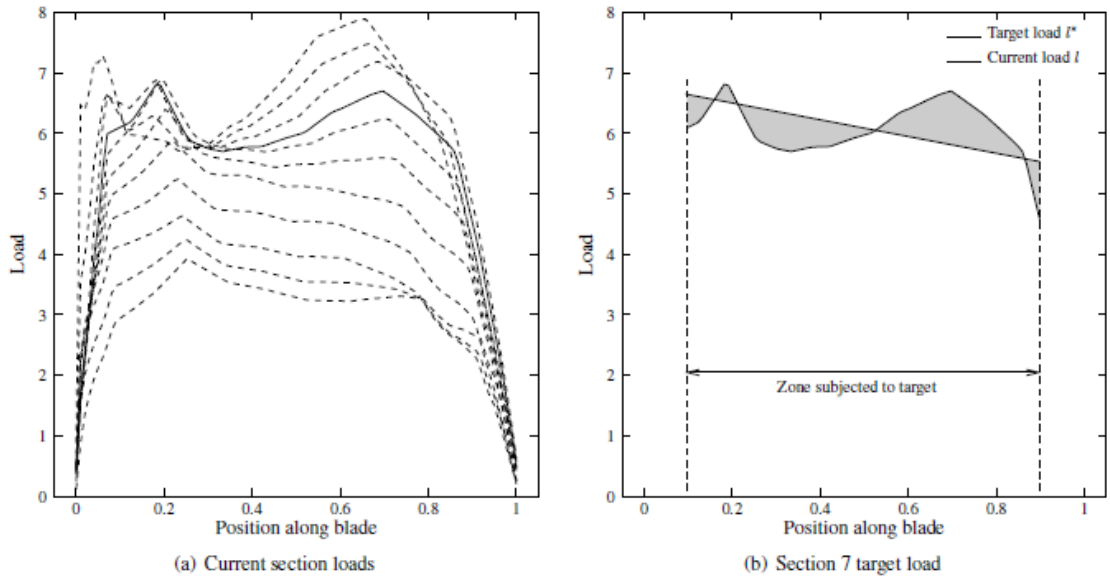


Figure 3-7 : Blade loading: (a) Loading curves of different blade sections. (b) Target load definition based on the section loading curve

The fourth constraint aims to prevent cavitation phenomena along the blade. The cavitation issue is addressed by controlling the minimum regional pressure on the blade (C_{Cav}). In order to do that, a limit dedicated to the minimum allowed pressure, U_{Cav}, has been defined for a specific blade region, which is the main candidate for the typical blade cavitation. This constraint has to be

evaluated at the full-load operating condition (i.e. OP_2). The other constraints and both objectives are evaluated at the expected BEP (i.e. OP_1).

Targets used in the objective/constraint formulations are fixed until the end of the optimization. However, the first operating point is corrected in each design step using the linear approach presented in Section 3.3.4.1 based on the difference between the BEP of the optimized design obtained from high-fidelity evaluations and the low-fidelity optimization. The relaxation factor is equal to one for the first correction. It is reduced to 0.8 in the second step of scenario B. Also the design space is modified in each step by adding new design variables.

3.5.2 Blade parameterization and design variables

Two different design optimization scenarios have been defined using a fixed total budget comprising 30,000 potential flow and 18 viscous flow evaluations. This computational balance is possible because the selected flow characteristics are well predicted by potential flow analyses in the current design process of hydraulic turbine runners. In both scenarios, the optimization starts with an initial model using 10 geometric design variables in the first design step, and ends in the last step with 17 design variables to have more flexibility (see Table 3.2).

Table 3.2 : Optimization scenarios

Scenario	Step	No. of potential flow evaluations	No. of viscous flow evaluations	No. of design variables
A	A1	15000	9	10
	A2	15000	9	17
B	B1	10000	6	10
	B2	10000	6	13
	B3	10000	6	17

Blade geometries are parameterized through an in-house software developed by Andritz Hydro Canada Inc. [109]. The software allows changing the number of parameters representing the geometry, modifying each of them, and visualizing the geometrical model. Curvature of the blade is governed by the beta angle parameters on nodes along streamline sections from the leading edge to the trailing edge. Delta length is a parameter which defines the variation of camber line length in each section. Local modifications are applied to delta beta, which means that all the existing nodes are allowed to be changed independently. Global modification is applied to delta length,

which means that this parameter varies globally and section lengths are not allowed to be changed independently, in order to obtain the same shape of the trailing edge in the R-Z plane. Thus, global modification of delta length adds one more design variable to the optimization problem (see Table 3.3). Delta beta and delta length parameters can vary from their initial value plus/minus upper/lower bound limits. At the beginning, delta beta bounds are determined based on the initial values of beta parameters and defined angular limits, which range from 0 to 90 degree. In the next optimization steps, those bounds are tightly limited to the new base geometry values (i.e. up to minus/plus 10%), mainly in order to keep major characteristics of the new base geometry by doing smaller detailed modifications, as well as preventing blade surface waviness by increasing the number of beta points. Delta length bounds are set from the initial average blade length minus/plus about 20%. If feasible optimization solutions are concentrated close to those bounds, they can be expanded in the next optimization step. A preliminary phase before optimizing consists in reducing the number of geometric parameters of the original geometry. During optimization, the overall dimensions of the runner and number of blades are fixed.

Table 3.3 : Independent design parameters and variables

	No. of parameters and variables			
	Initial model (original)	Initial model of A1 & B1	Initial model of B2	Initial model of A2 & B3
Blade curvature	25	9	12	16
Blade length	8	1	1	1
Blade thickness	165	Fixed	Fixed	Fixed
Blade leading edge	11	Fixed	Fixed	Fixed
Crown/band contours	40	Fixed	Fixed	Fixed
Number of blades	15	Fixed	Fixed	Fixed

3.5.3 Results and discussions

In each design scenario 30,000 potential flow analyses took about 15 hours using an Intel Core i7-2600 CPU at 3.4 GHz. In the high-fidelity phase, the corresponding computation time for viscous flow analyses of 18 selected candidates was 57 hours, using eight parallel processing units of an IBM X series server with four Xeon X7550 2 GHz CPUs. It is 79% of the overall computational time spent in this case study. The computational time of the filtering process was negligible.

Optimization results are summarized in Table 3.4. In the high-fidelity phase, BEP of selected design candidates has been compared with the original design performance at the new operating point. The deviations of BEP from the targeted operating point have been presented in this table as well. Although the efficiency improvement has not been changed significantly in the new steps of optimization, but new designs are better because BEPs are closer to the targeted operating point. According to Section 2.4, it demonstrates that the corrected low-fidelity optimization problems are adequately good representative of the high-fidelity design problem.

In scenario A and B, 9521 and 9132 feasible solutions were obtained respectively, which are 32% and 30% of all optimization evaluations. In the low-fidelity phase, the constraint restrictions and geometric flexibility play an important role in reaching enough feasible solutions. For instance, Figure 3-8 illustrates tangential velocity objective improvements of feasible solutions. It is quite similar for the other objective too. In the first optimization steps (A1 and B1), the major improvement was achieved in the first 1000 iterations; after 4000 iterations the objective value improvement was relatively negligible, while there were enough feasible solutions. B2 started with a jump, which is due to modifying the objective operating point. Although this jump is smaller for A2, even with 15000 optimization iteration, A2 has led to a higher objective value (i.e. less improvement).

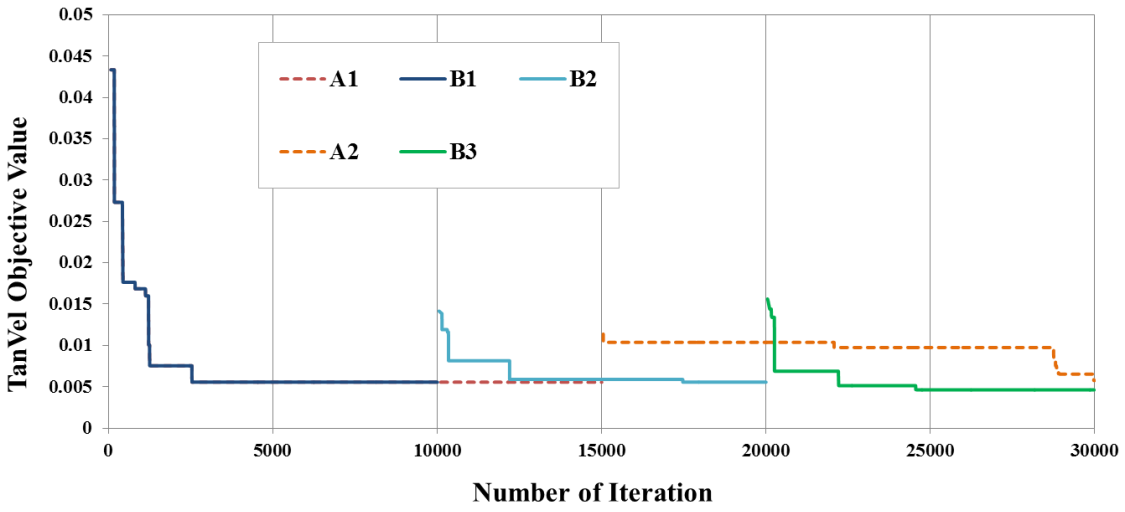


Figure 3-8 : Improvement history of tangential velocity objective function using two optimization scenarios

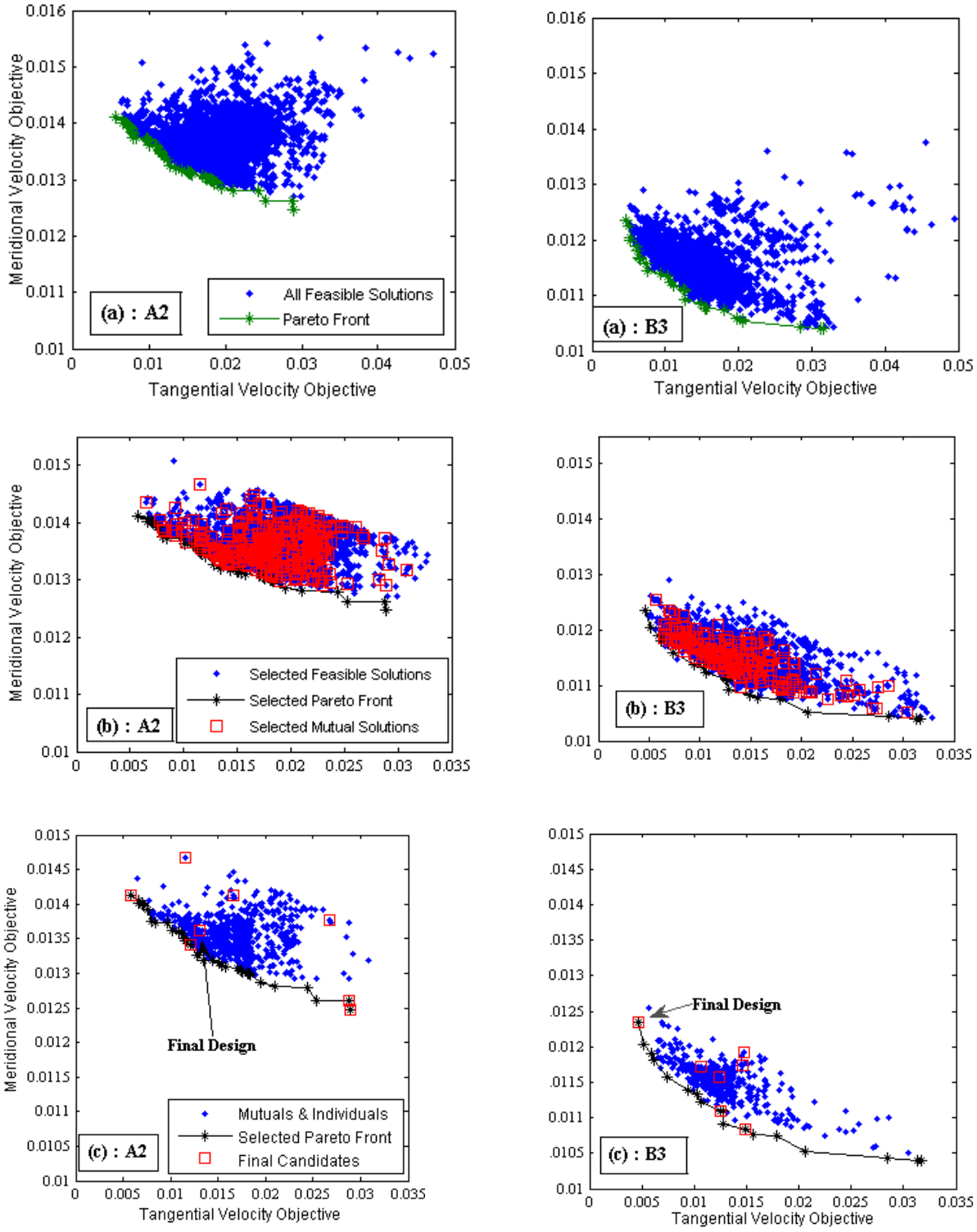


Figure 3-9 : (a) Optimization results of A2 and B3 in the objective space. (b) Selected solutions after sieving process. (c) Selected candidates after clustering

Figure 3-9 shows two examples of the filtering process associated with the last step of the two scenarios (i.e. A2 and B3). Part “a” illustrates all feasible solutions among all evaluated optimization solutions. Part “b” shows the selected dissimilar candidates using the sieving process. In part “c”, nine and six selected candidates are shown respectively, which were chosen among candidates selected in part “b” via clustering. Also, the final designs are shown, which were determined after viscous evaluations of the selected candidates in the high-fidelity phase. As part “c” indicates, the best design candidate is not necessarily among Pareto members; it can be captured outside the Pareto front as a benefit of the filtering process (e.g. A2).

The best design was determined by comparison of two final designs obtained in A2 and B3, which have been shown in Figure 3-9 (c). According to Table 3.4, scenario B has been more successful to reach higher efficiency at the right operating point. It also indicates that in scenario B, selecting a fewer number of candidates, six candidates in each step (33% fewer), has not caused significant drawbacks.

Figure 3-10 (a) shows the turbine efficiency curves for the original and two final optimized runners. In this figure, efficiencies and power coefficients have been normalized respectively by the BEP efficiency and the power coefficient of the original runner. Figure 3-10 (a) illustrates that efficiency curves of A2 and B3 designs have been shifted close to the expected operating point (BEP at normalized power coefficient=1.12), with 2.86% and 1.09% lower power coefficients respectively. Both optimized designs also have led to an efficiency improvement. It indicates the ability of the proposed design optimization algorithm in terms of the efficiency (since the original blade was an efficient existing blade) as well as reaching the BEP target, in only a few (three) design steps. The improvement of the efficiency is a result of decreasing total losses of turbine components. Figure 3-10 (b) shows the comparisons of runner/draft tube losses, as the most important components influenced by the proposed design optimization system, and dictate the efficiency curve. Scenario B resulted in a more efficient runner with minimum runner losses, and minimum draft tube losses close to the expected operating condition.

Table 3.4 : Optimization results

	Low-fidelity phase; Optimization results		**High-fidelity phase; BEP of the best of selected candidates**	
	No. of evaluations	No. of feasible solutions	Efficiency improvement %	Deviation from targeted *OP %*
A1	15,000	5143	1.59	-3.0
A2	15,000	4378	1.61	-2.86
B1	10,000	3266	1.60	-3.29
B2	10,000	2481	1.66	2.50
B3	10,000	3385	1.64	- 1.09

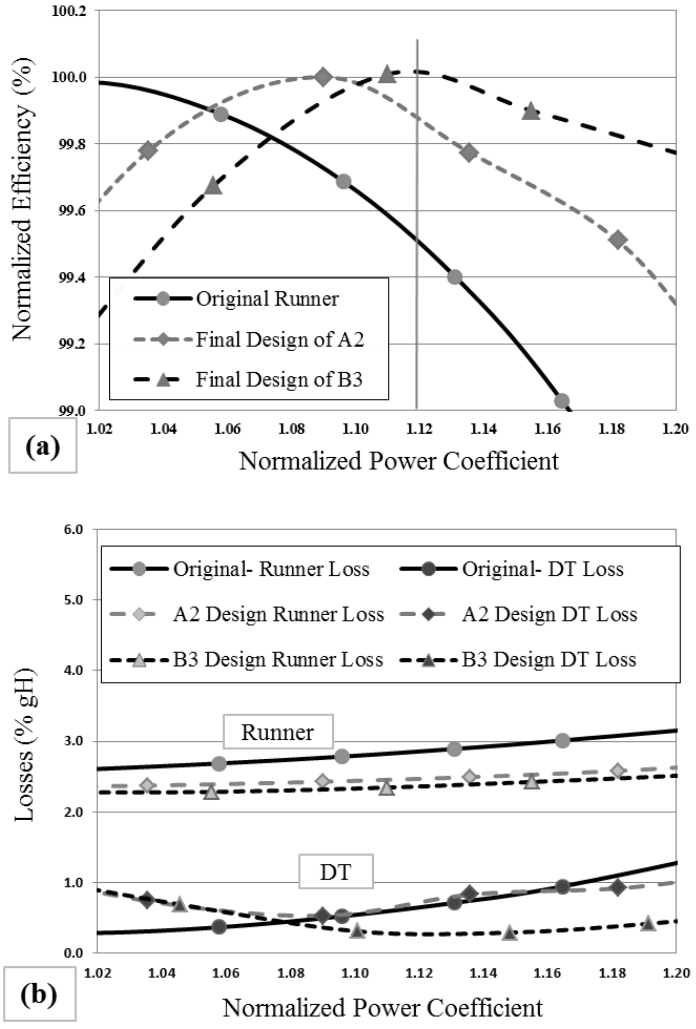
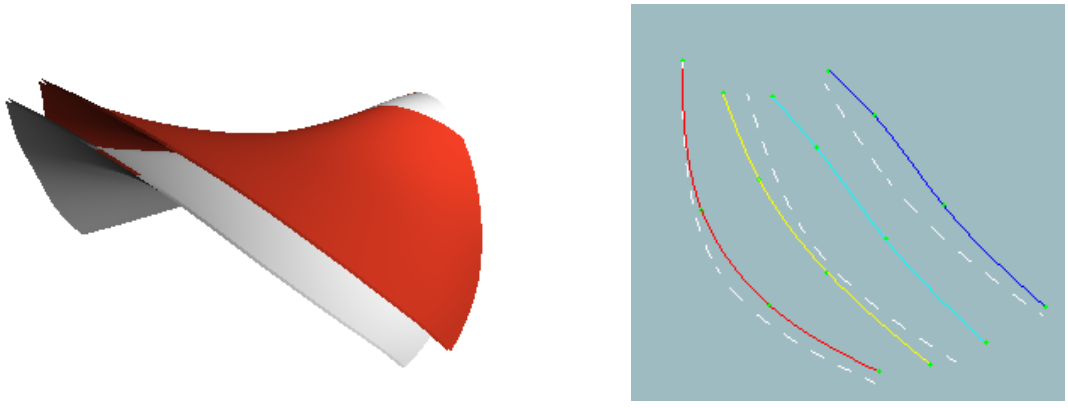


Figure 3-10 : (a) Efficiency curves. (b) Losses of runner and draft tube

Figure 3-11 (a) illustrates the geometry differences between the original and the B3 optimized blade. The optimized blade is approximately as long as the original one. Figure 3-11 (b) shows a comparison of the curvature of the original blade represented by nine points and the optimized blade represented by 16 points. The blade curvature modifications play the main role to control the runner outlet velocity profile. Outlet tangential velocity profile of the initial geometry has been changed significantly to approach the defined target, which was a flat zero target at selected central flow length (see Figure 3-12). Also, high-fidelity evaluation results have indicated good predictions of velocity profiles obtained in the low-fidelity phase by the potential flow solver. This small gap between low- and high-fidelity results validates usage of the proposed methodology and the choice of characteristic, which can be predicted accurately enough and be used in an optimization objective function. However, this gap cannot be considerably reduced since the low-fidelity analysis (i.e. potential flow solver) is not correctable.

On the optimized blade, proper pressure distribution causes a relatively uniform load distribution. It also leads to the elimination of partial pressure drop and to an increase of the minimum pressure on the blade. These achievements are mainly due to a proper definition of load distribution and pressure constraints in the low-fidelity optimization. For instance, Figure 3-13 shows considerable improvements in smoothness of the blade loading, which presents a front-loaded distribution with a smooth load variation from crown to band as well as from leading edge to trailing edge.



(a) (b)

Figure 3-11 : (a) Geometry comparison (white: the original blade, red: B3 optimized blade), (b) Curvature comparison (original blade represented by dashed line)

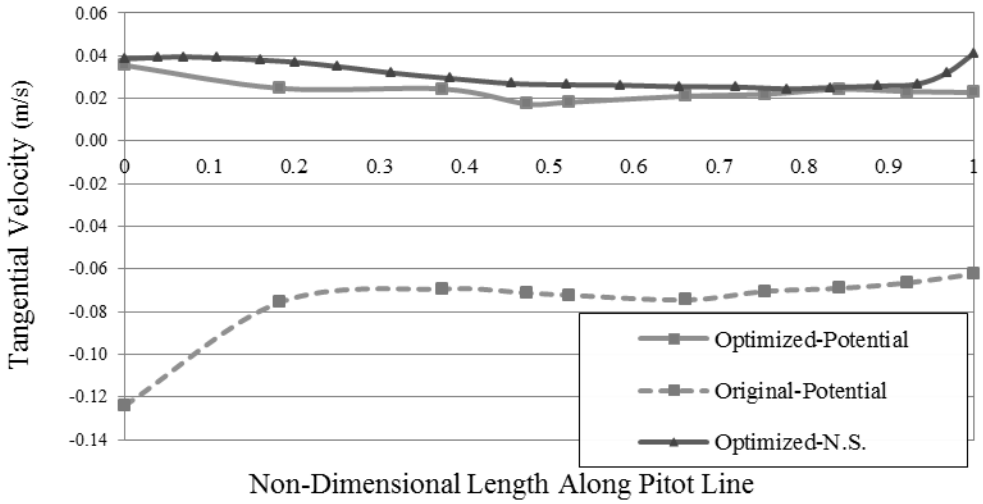


Figure 3-12 : Tangential velocity improvement of B3 optimized blade at the targeted BEP

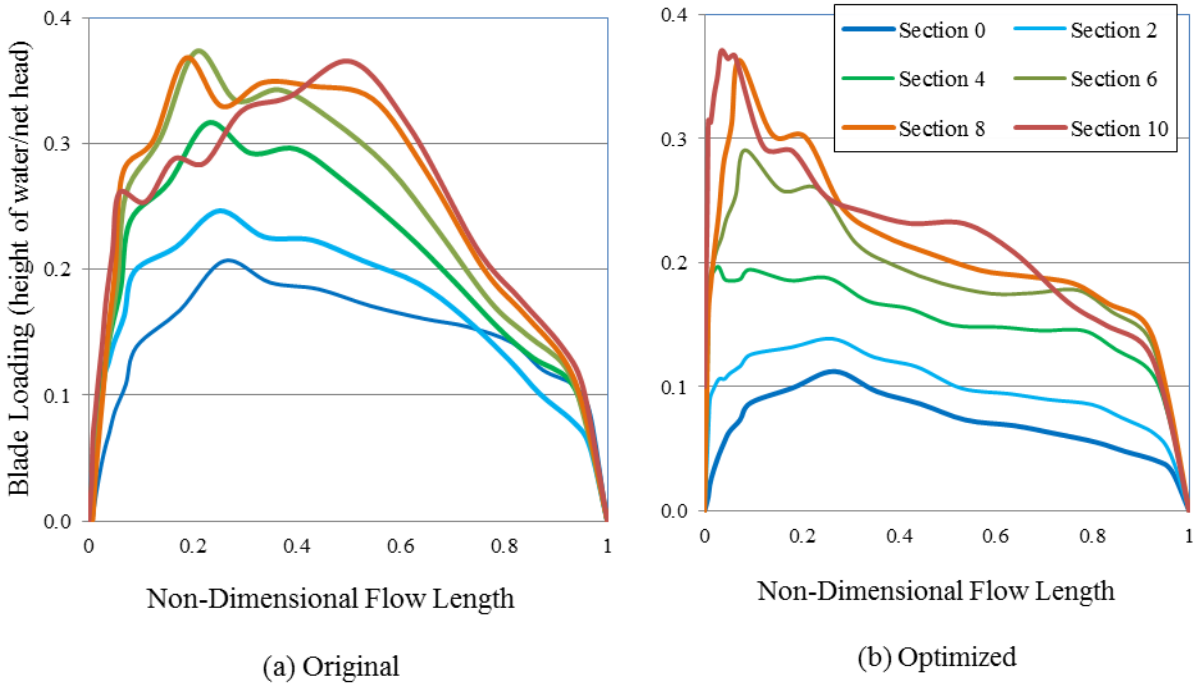


Figure 3-13 : Blade loading comparison between (a) original blade and (b) B3 optimized blade at new BEP

3.6 Conclusion

In this work, a new robust multi-fidelity design methodology has been developed, using a physics-based surrogate optimization, which brings the big advantage of significant evaluation cost/time reduction. It aims to split design evaluations between low- and high-fidelity phases in order to properly balance the evaluation cost and required accuracy in different optimization stages. This methodology addresses those types of multi-phase problems where the high-level objectives and/or constraints are not directly assessable in the low-level, which is a challenge of some industrial problems in different fields. This challenge has not been investigated properly in the literature. As another contribution, two mechanisms are defined to improve the low-fidelity optimization: target correction and design variable refinement. They are quite important, since unlike the functional surrogate, physics-based surrogate accuracy cannot be improved. While the first mechanism recalibrates the optimization features, the second one prepares higher-fidelity geometry to provide more chance of objective/constraint improvements.

In the low-fidelity phase, a derivative-free optimization method employs inexpensive low-fidelity analyses to obtain the major desired characteristics of a good design in a relatively fast iterative process. After completing the low-fidelity phase, a limited number of candidates are selected by a newly developed filtering process to be sent to the high-fidelity phase. The developed filtering algorithm takes advantages of simple methods that make it computationally cheap, robust, and easy to implement. The high-fidelity phase is in charge of accurate assessment of filtered candidates to select the best one. It is also responsible for recalibration of the low-fidelity optimization by correcting objectives and constraints.

The developed methodology was applied to redesign a Francis turbine runner to meet a higher power output, within two design scenarios with a fixed computational budget. Optimization was started from a low-fidelity parameterized model (10 parameters), and ended with a higher-fidelity model (17 parameters). The scenario with more global design steps led to a better final design which brought better efficiency of the turbine, and also had the expected BEP location. It is concluded that it is more effective to split the given computational budget among more design steps with fewer evaluations in each step, mainly in order to tune the low-fidelity phase more often by high-fidelity information. The numerical design process took 72 hours to reach the final optimized blade. High-fidelity analyses contained the majority of the budget (i.e. 79%), which can be

significantly reduced by employing more powerful computational resources as well as more efficient high-fidelity computational algorithm.

The reported results of similar blade optimization in the literature were obtained by high-fidelity analyses of new geometries, using a few hundreds up to several thousand analyses, yielding very significant optimization time; while, in this investigation, much larger numbers of new geometries were evaluated in each scenario, and the optimized blade was obtained in significantly lower computational efforts. It also should be noted that 30,000 geometry evaluations in each scenario with a remarkable low computational burden was a big advantage, since the blade geometry optimization is quite a noisy problem and better design space exploration is a key. Thus, the developed multi-fidelity methodology is a good choice for most complex industrial optimization problems. It can be applied efficiently in different design cases by some case adaptations.

In the future, more investigations will be performed to improve the automation of the proposed methodology. In the hydraulic turbine field, the effect of other design variables (e.g. contours and leading edge) and their priorities are going to be determined. Also, other effective parameters should be studied to define appropriate objectives and constraints for the low-fidelity optimization, to address directly or indirectly other important aspects of a good design in the high-fidelity phase, such as the flatness of the efficiency curve.

3.7 Acknowledgment

The authors would like to thank Andritz Hydro Canada Inc. and NSERC, Natural Sciences and Engineering Research Council of Canada, for supporting this project.

CHAPTER 4 ARTICLE 2: PHYSICS-BASED SURROGATE OPTIMIZATION OF FRANCIS TURBINE RUNNER BLADES, USING MESH ADAPTIVE DIRECT SEARCH AND EVOLUTIONARY ALGORITHMS

S. Bahrami[1], C. Tribes[1], S. von Fellenberg[2] , T. C. Vu[2] and F. Guibault[3]

[1] *Mechanical Engineering Department, École Polytechnique de Montréal, Montréal, Quebec, Canada, H3T 1J4*

[2] *R&D Division, Andritz Hydro Canada Inc., 6100 TransCanada highway, Point-Claire, Quebec, Canada, H9R 1B9*

[3] *Computer Engineering Department, École Polytechnique de Montréal, Montréal, Quebec, Canada, H3T 1J4*

*Based on the paper published in the "International Journal of Fluid Machinery and Systems", Vol. 8, No. 3, pp. 209-219, 2015.

4.1 Abstract

A robust multi-fidelity optimization methodology has been developed, focusing on efficiently handling industrial runner design of hydraulic Francis turbines. The computational task is split between low- and high-fidelity phases in order to properly balance the CFD cost and required accuracy in different design stages. In the low-fidelity phase, a physics-based surrogate optimization loop manages a large number of iterative optimization evaluations. Two derivative-free optimization methods use an inviscid flow solver as a physics-based surrogate to obtain the main characteristics of a good design in a relatively fast iterative process. The case study of a runner design for a low-head Francis turbine indicates advantages of integrating two derivative-free optimization algorithms with different local- and global search capabilities.

Keywords: Physics-based surrogate optimization, Francis turbine runner blade, multi-fidelity algorithm.

4.2 Hydraulic turbine design optimization process

Big changes in global energy demand, increasing environmental concerns, and growth potential of cost-efficient hydroelectric energy, have recently resulted in more demand to design hydraulic

turbines which are more efficient and durable. As design challenges are getting more complex, runner designers rely more than ever on engineering and simulation tools, specially computational fluid dynamics (CFD), to obtain reliable designs with a competitive time and cost. Although runner designers already employ CFD tools to evaluate their designs, there is a strong need to integrate CFD analyses more tightly in the design chain using efficient optimization methods to obtain more efficient design processes.

The full range of CFD methods have been utilized in the optimization of hydraulic turbine runner blades. Low-fidelity inviscid models (e.g. potential flow) have been employed by some researches such as Holmes and McNabb [17]. However, they are not accurate enough in their prediction of flow behavior, mainly due to lack of physics. High-fidelity viscous models have been used alone to optimize the runner as well (e.g. using turbulent RANS solvers, by Franco-Nava et al. [110] and Pilev et al. [111]); but they are too expensive and slow for iterative industrial runner design processes. To reduce high-fidelity analyses in the optimization loop, surrogate-based optimization approaches have been increasingly employed by researchers, using either mathematical surrogates or physic-based surrogates. Mathematical surrogates are computationally inexpensive approximation models constructed from a given number of high-fidelity evaluations. For instance, artificial neural network was applied by Derakhshan et al. [112] to reduce Navier-Stokes solver calls during the optimization of a low-head axial hydro turbine by an evolutionary algorithm. Another popular mathematical surrogate, radial basis functions, was employed by Georgopoulou et al. [63].

Although those mathematical surrogates have been used for blade shape optimizations, they still require a large number of high-fidelity viscous evaluations to update and to ensure that they yield reasonably accurate results. For this reason, physics-based surrogate can be a better alternative in some situations, which uses simplified physics of the problem (i.e. low-fidelity evaluations). Physics-based surrogates are usually used in multi-fidelity frameworks mainly in order to correct low-fidelity evaluations of objectives and constraints using high-fidelity results.

Beside all the aforementioned researches, industrial runner design process currently relies extensively on the designer’s intuition and experience, using both inviscid and viscous flow analyses, but mostly without using an optimizer. Runner designers can use fast inviscid flow solvers, to carry out most design iterations in early phase of the design process. The high-fidelity

analyses are also considered to further assess the design quality. To fulfil industrial design needs, a new practical multi-fidelity methodology has been developed using a physics-based surrogate optimization in the low-fidelity phase. In this methodology the optimizer employs a computationally-cheap inviscid flow solver in the low-fidelity phase. The low-fidelity optimization problem is corrected by accurate high-fidelity information in an overall design loop. Previous investigations (e.g. by Alexandrov et al. [89], Robinson et al. [88], and by Leifsson and Koziel [49]) indicate that multi-fidelity methods require much fewer high-fidelity evaluations than mathematical surrogates to obtain a given level of accuracy. To comply with runner designers' approaches, the proposed design framework involves all existing design resources (see Figure 4-1), adding an automatic optimization loop to decrease designer interactions.

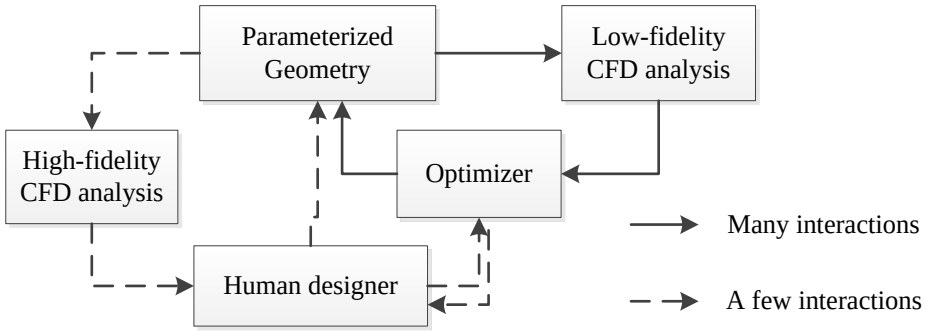


Figure 4-1 : Runner design loop interactions

In the next section, the multi-fidelity optimization methodology is presented. In section 3, a case study is presented using a low-head Francis runner. Even though we presented the multi-fidelity methodology using the same test case in 27th IAHR Symposium on Hydraulic Machinery and Systems [113], the current enhanced paper focuses on how the exploration capabilities of optimization methods affect the optimization efficiency and robustness to obtain good Francis runner designs. Mesh adaptive direct search (MADS) and evolutionary algorithm have been employed, which are mostly well-known as good local- and global search methods respectively. Among different types of derivative-free optimization methods, these two methods have demonstrated their abilities in hydraulic optimization problems.

4.3 Multi-fidelity design optimization methodology

4.3.1 Low-fidelity phase

The main iterative computations are carried out in the low-fidelity phase which contains the low-fidelity optimization loop (see Figure 4-2). It aims to approach the main design characteristics via thousands of fast and computationally inexpensive low-fidelity (i.e. inviscid flow) evaluations within an optimization loop. This loop starts with a parameterized model using a few design variables representing the initial geometry. Inviscid flow field calculations produce the required information, such as velocity and pressure distributions on the blade, to evaluate objective functions and constraints. The optimizer determines new design variable values, based on the improvement or deterioration in the objective and constraint values. Since inviscid flow solvers cannot consider viscous effects, the low-fidelity optimization phase focuses on meeting specific target flow characteristics that are indirectly associated with low energy losses (equivalently high machine efficiency) and cavitation absence at the given operating conditions. In this project, a potential flow solver has been chosen as the physics-based surrogate.

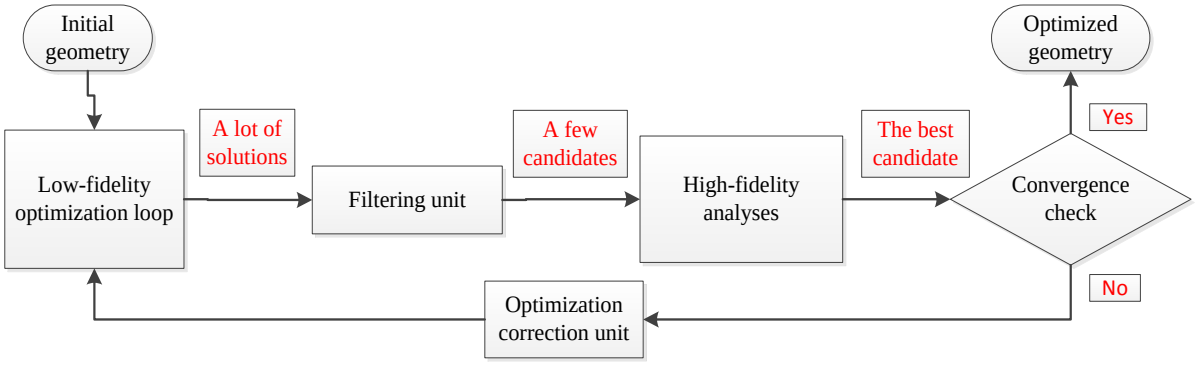


Figure 4-2 : Multi-fidelity design optimization algorithm

In this investigation, the low-fidelity evaluations cannot be used to determine derivative information with enough accuracy. Moreover, the complexity of the design space and presence of many local optimums requires both local and global design space explorations. For these reasons, we have selected two different derivative-free optimization methods. NOMAD (Non-smooth Optimization by Mesh Adaptive Direct Search) [93] has been selected as an open source [94] C++ implementation of the Mesh Adaptive Direct Search algorithm [95]. Also, a well-known evolutionary algorithm has been used by employing an optimization code, evolutionary algorithm

system (EASY) [114]. It has been used in similar investigations, such as Francis runner and radial pump impeller optimization [115] and draft tube optimization [116]. The details of this optimizer and evolutionary algorithm can be found in [57].

4.3.2 Filtering process

Based on the number of objectives, the dominant solution or a set of them (i.e. Pareto optimal solutions) can be identified after the low-fidelity optimization. However, differences between low-fidelity optimization results and high-fidelity Navier-Stokes results are expected, which are due to assumptions made through the use of inviscid flow evaluations. Preliminary investigations have shown that other feasible optimization solutions, not too far from dominant solutions, can also bring high efficiency in Navier-Stokes evaluations. Therefore, a versatile filtering algorithm (filtering unit in Figure 4-2) has been developed to select a few promising candidates which are geometrically different and dominant in their own territories. These candidates are transferred to the high-fidelity phase for Navier-Stokes evaluations.

The filtering process contains the following parts:

- Filtering feasible optimization solutions.
- Mapping the design space into a distributing standard hypercube.
- Distributing those solutions into small unit-length hypercubes.
- Selecting one dominant candidate from each unit-length hypercube.
- Selecting a few geometrically-different candidates out of all selected candidates via a clustering method.

The details of proposed filtering process are available in [92].

4.3.3 High-fidelity phase

In the high-fidelity phase, a viscous flow solver is used to accurately evaluate a few selected candidates. It aims to choose the best design candidate which has a good efficiency at the right operating condition, and minimum cavitation. This design candidate may be transferred into the low-fidelity phase as a new initial design for the next optimization step, or selected as the final design based on certain convergence criteria or computational budget limit. The number of design

variables of the best candidate may be increased when it is transferred, mainly in order to give more flexibility to the optimizer to satisfy real constraints in the next step. By analyzing the results of the high-fidelity phase, it is possible to recalibrate the objectives and constraints, in order to obtain the desired results and expected final goals in the high-fidelity phase.

The commercial Navier-Stokes code ANSYS-CFX has been employed for the flow field simulations using the standard two-equation k-ε RANS turbulence model. Due to rotational periodic conditions, the viscous flow analysis is performed only for a single passage of the runner flow. This domain is discretized using approximately 200,000 structured cells. Details of the methodology, integrated tools and validations can be found in the references [105-107].

4.4 Low-fidelity optimization arrangement

All multi-fidelity optimization formulations including the low-fidelity optimization problem have been presented extensively in [117]. The low-fidelity optimization problem is formulated as:

Minimizing $f_i(y)$

Subject to $g_i(y) \leq 0$; $y_l \leq y \leq y_u$

Where "y" is the N-dimensional vector of design variables, and "y_l" and "y_u" are respectively the lower and upper bounds of design variables. One objective and three constraints have been defined in this project.

4.4.1 Objective and constraints

Previous investigations have shown that minimizing the length of the blade is a good way to drive the optimization towards good runner geometries. The objective function is calculated from summation of weighted section lengths.

Different types of constraints may be used in hydraulic runner optimization. For this investigation, three constraints control the most important design criteria addressing minimum losses and maximum efficiency at the targeted operating condition:

- Velocity constraint: To control the losses in the draft tube, it is necessary that the runner delivers an appropriate tangential velocity profile at the draft-tube inlet [10, 30]. Therefore,

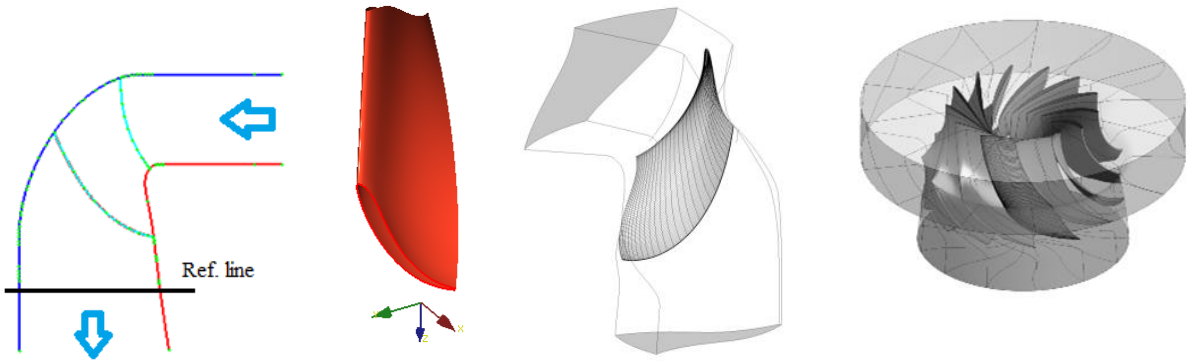
this velocity component at the runner outlet reference line has been determined to be similar to a targeted profile within a safe bound, which is based on designers' experiences.

- Blade loading: one constraint prevents negative blade loading on all blade sections.
- Cavitation: one constraint has been defined to limit the minimum allowed pressure for all blade sections, in order to represent cavitation issues during the optimization.

4.4.2 Initial geometry and design variables

A low-head Francis turbine runner has been chosen. The goal is to design a runner such that on a given efficiency curve (speed coefficient Ned equal to 0.407) it provides the peak position at a given flow coefficient P_{ed} equal to 0.294. This turbine contains 13 runner blades and 20 guide vanes. The initial geometry is a poor hydraulic blade shape.

Figure 4-3(a) shows blade edges and runner inner/outer contours, projected in the meridional plane. The tangential velocity reference line has been shown as well. Figure 4-3(b) illustrates the simple geometry of the initial blade. Figure 4-3(c) shows the aggregated fluid flow domain considered in the computational analyses. Inlet and outlet boundaries are illustrated on the top and bottom of the blade respectively. Figure 4-3(d) illustrates the whole runner considering rotational periodicity of the flow domain. The blade thickness has been created using a NACA blade thickness profile (see Figure 4-4).



(a) (b) (c) (d)

Figure 4-3 : Initial blade geometry, single-blade and runner flow computational domain

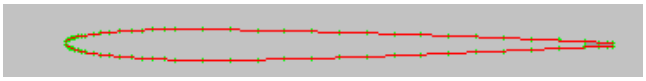


Figure 4-4 : Blade thickness profile

Blade geometries are parameterized through an in-house software developed by Andritz Hydro Canada Inc. [109]. The software allows changing the number of parameters representing the geometry, modifying each of them, and visualizing the geometrical model.

Table 4.1 shows the main geometric parameters and design variables. Curvature of the blade is governed by some angle control points along streamline sections from the leading edge to the trailing edge. The lean of the leading edge is represented by other angle points on a 2D projected curve from band to crown. Blade length is defined by the camber line length in each section. The band-side contour is optimized using cylindrical coordinates (i.e. r and z) of two points located downstream of the leading edge on the band. Proper bounds of variations are defined for each four types of design variables, based on some geometric limits and/or designer experiences.

Table 4.1 : Number of independent parameters & design variables

	Initial model parameters	**Optimization design variables**
Blade curvature	9	9
Blade length	11	3
Blade leading edge	11	1
Band contour	20	4
Crown contour	43	Fixed
Blade thickness	80	Fixed
Number of blades	13	Fixed

4.4.3 Optimization features

A maximum number of 40,000 low-fidelity evaluations have been set for both optimizers. Also, the same problem formulation has been used for both optimizers.

Although only one objective is used for the problem at hand, it should be noted that both optimizers are able to handle multi-objective optimization problems. For instance, we presented the results of a bi-objective low-fidelity optimization employed in the developed multi-fidelity methodology [92]. Also, EASY has demonstrated its success to handle a multi-objective Francis runner optimization problem [115].

4.4.3.1 NOMAD

Since NOMAD is a mesh adaptive method, it will stop when reaching the mesh convergence criterion even if the maximum number of evaluations is not used. NOMAD needs an initial design vector to start exploring the design space. Our previous investigations have indicated that in blade shape optimization problems, NOMAD performance and optimization results are quite sensitive to the initial design vector. In order to alleviate this drawback, a Latin Hypercube (LH) method has been used with 1% of the maximum computational budget, to evaluate 400 new design points at the beginning, and select the best one to obtain the most promising initial vector.

Another important NOMAD aspect is the assignment of global search budget, which has been investigated in this study as well. Although the MADS algorithm is quite powerful in local search, it needs some special considerations to be successful in global search too. Variable Neighborhood Search (VNS) is an algorithm integrated into NOMAD to improve global search of the design space [50]. While the default value of VNS budget is equal to 75% of overall budget, two more global search budgets have been also investigated using VNS equal to 85% and 95% of the overall budget. The importance of global search is intensified in the problem at hand while the blade performance is strongly sensitive to small design variable changes.

All the constraints are considered as relaxable using the Progressive Barrier (PB) approach of NOMAD. With this approach, the MADS algorithm identifies new incumbent solutions by considering feasible points with the lowest objective values that improve feasibility. Optimization improvement is determined based on filter method of Fletcher and Leyffer [118].

4.4.3.2 EASY

A considerable number of evolutionary algorithm parameters have to be taken in to consideration in EASY. However, most of them have well-tuned default values. Based on previous investigations, the length of chromosomes (strings of binary digits representing values of design variables) and the population size (number of offspring generated in each iteration of evolutionary algorithm) have been chosen to be studied. Mutation- and cross recombination probabilities have been set to 0.02 and 0.9 respectively. Before each new geometry evaluation, EASY checks the database containing previous evaluated geometries to prevent repeating the same evaluation.

The evolutionary algorithms employed in EASY uses penalty functions to enforce constraints. For the problem at hand, proper constraint weights have been investigated and determined based on the ranges of infeasible constraints- and objective values.

4.5 Results and discussions

For the problem at hand with the selected bounds of design variables, the lowest possible objective value is 1. Table 4.2 shows the results of the optimization problem defined in the last section using NOMAD and EASY. Feasible solutions are selected in the first step of the filtering process. As it was expected, EASY has achieved much larger numbers of feasible solutions due to its exceptional global search capacity. However, generally NOMAD is more capable in local search since it almost always has obtained the best possible objective value faster.

Table 4.2 : NOMAD- and EASY-based optimization performances

			No. of Eval.	**No. of feasible solutions**	**First feasible solution**	**Obj. value of the first feasible**	**Best Obj. value**
NOMAD	VNS 0.75	X0	25679	0	-	-	-
		LH	15945	1460 (9.2%)	7981	1.04	1.00
	VNS 0.85	X0	37754	4175 (11.1%)	27191	1.14	1.00
		LH	27699	2494 (9.0%)	19039	1.07	1.00
	VNS 0.95	X0	40,000	1163 (2.9%)	5063	1.43	1.28
		LH	40,000	1412 (3.5%)	9176	1.09	1.00
EASY	P 50	L 5	40,000	25558 (63.9%)	1839	1.23	1.01
		L 10	40,000	25789 (64.4%)	1512	1.22	1.00
		L 15	40,000	10997 (27.5%)	2007	1.20	1.10
		L 20	40,000	23698 (59.2%)	1587	1.22	1.00
	P 30	L 10	40,000	0	-	-	-
	P 40	L 10	40,000	0	-	-	-
	P 60	L 10	40,000	15246 (38.1%)	8984	1.25	1.20
	P 70	L 10	40,000	0	-	-	-

Six NOMAD optimization results indicate LH search at the beginning of the optimization is really helpful, especially in lower VNS budgets. For instance, while using an initial design vector (i.e. X0) has led to no feasible solution, using LH to find a relatively good initial vector has improved extensively the performance and allowed achieving 1460 feasible solutions in 38% fewer number of evaluations (due to NOMAD mesh convergence stopping criterion). In this optimization the objective value of the first feasible solution was quite good which caused reaching the best possible

objective value very quickly. The aforementioned point has been well demonstrated in Figure 4-5. The bigger the VNS budget assigned, the more chance of globally exploring the design space and getting away from the local minima in order to find other good solutions. Disconnected curves in Figure 4-5 (e.g. dedicated as VNS 0.95_LH) indicates that point, since each of those disconnections shows the VNS effect by stopping local search and jumping out of the previous search regions, which usually causes starting from new infeasible regions. This figure also indicates that NOMAD is really sensitive to the initial design vector and confirms the necessity of LH usage. For instance, using the highest VNS budget (i.e. 0.95) without LH, could not improve the relatively big objective value of the first feasible solutions. Although in this optimization the first feasible solution was obtained quite fast (at the 5063rd evaluation), completing all 40,000 evaluations did not lead to a better objective value than 1.28. Among NOMAD optimizations, VNS 0.85_LH has been chosen for further investigation since it has explored the design space with a medium global search budget with a relatively low number of evaluations due to quick local search convergence.

EASY optimization results indicate that it is really sensitive to the studied parameters, especially to the size of population. Thus, it should be carefully calibrated at the beginning. The best possible objective value has been achieved only by using the population of 50. No feasible solution has been obtained using population sizes of 30, 40 and 70. The results also indicate that among different chromosome lengths, lengths of 10 and 20 have resulted in the best objective value. A larger number of feasible solutions and a better convergence are achieved using length of 10 (see Figure 4-6). Therefore, it was selected for the investigation of population size effect. Also, the case with population size of 50 and length of 10 (called P50_ L10) has been chosen for further investigation.

Global search capability of EASY can be illustrated by investigating the distribution of feasible solutions in the design space. For instance, Figure 4-7 shows that distribution for two selected optimization results of NOMAD (VNS 0.85_LH) and EASY (P50_L10) from six design variable points of view. In all of them, EASY has covered significantly larger feasible regions. It is really important from the filtering point of view, since the main task of the filtering unit (see Figure 4-2) is selecting a certain number of promising candidates while they are geometrically as different as possible. Figure 4-7 indicates that among the presented design variables, there is a big concentration of feasible solutions on the lower bounds of length design variables; thus, optimization improvement may be reached by decreasing those lower bounds. It was expected in

advance since the objective function has been formulated as a summation of weighted length variables. However, as it was mentioned earlier, the lower bounds of length variables has been defined by experienced designers due to other design considerations, and consequently cannot be changed.

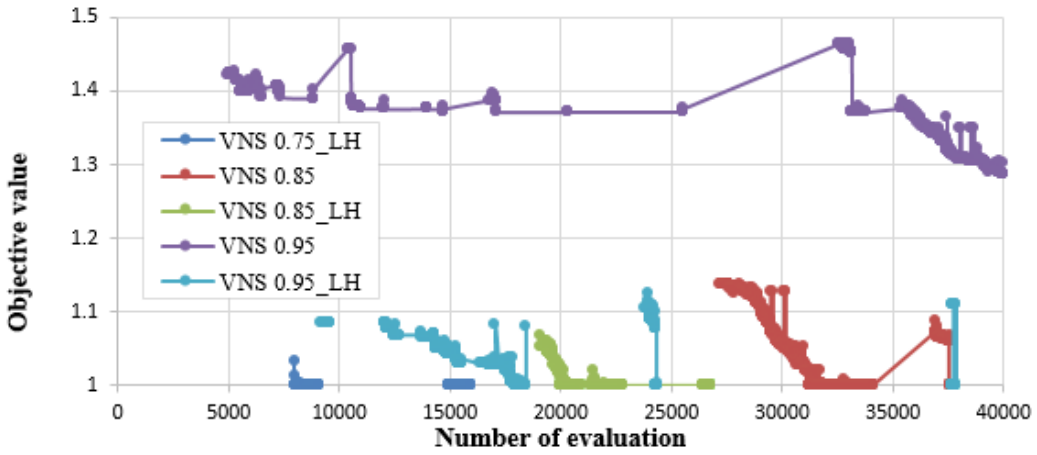


Figure 4-5 : Objective value improvement of feasible solutions obtained by NOMAD using different VNS budgets

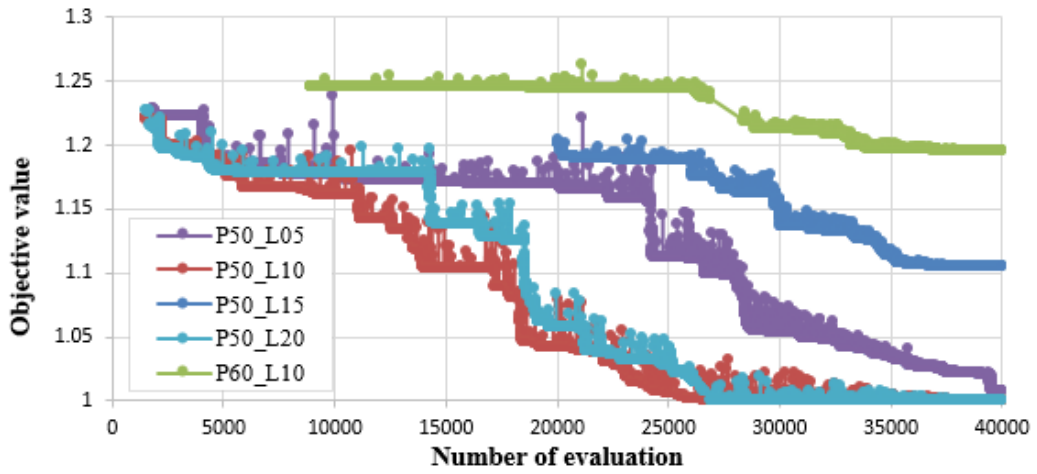


Figure 4-6 : Objective value improvement of feasible solutions obtained by EASY using different population sizes and chromosomes lengths

Figure 4-8 illustrates comparison of two selected optimization results with filtered candidates. As a good local optimizer, NOMAD converges quickly down to the best objective value once it achieves the first feasible solutions, although this achievement happens relatively late. However, EASY as a good global optimizer obtains a much larger number of feasible solutions from very

earlier evaluations, but it improves the objective value gradually within some major recognizable steps. Four promising candidates were chosen by the filtering unit for each of those two optimizations to be sent to high-fidelity phase.

Table 4.3 shows the results of High-fidelity Navier-Stokes analyses of the selected candidates. It indicates that the 21988th solution of NOMAD optimization is the best selected candidate, which has the improved efficiency curve peak at the targeted power coefficient within the allowed error range (±1% error for the problem at hand), while it has no cavitating area. If none of those candidates has the peak position within the range, the candidate with the best efficiency (e.g. 22705) may be chosen to be used as the new initial geometry for the next optimization step with corrected operating conditions. However, different selection policies may be applied for different situations.

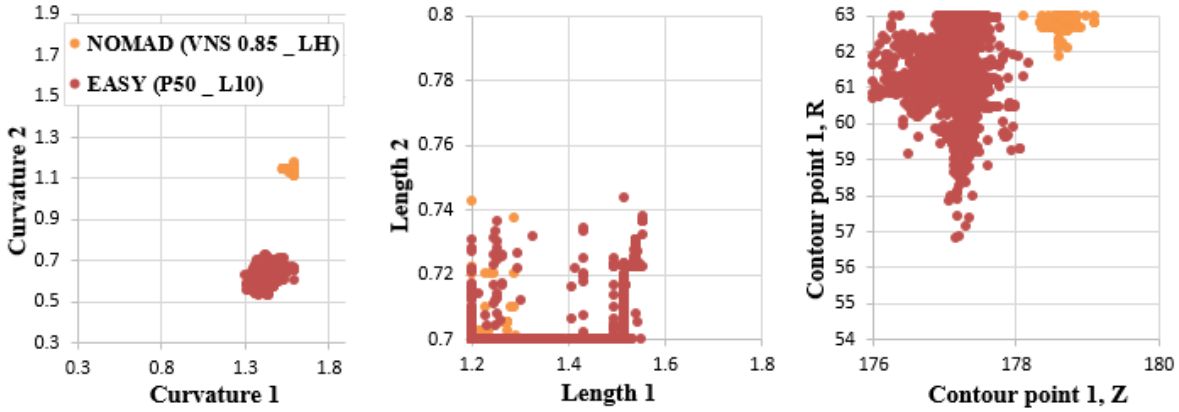


Figure 4-7 : Comparison of feasible solution distributions in the design space

Table 4.3 : High-fidelity evaluation results of filtered candidates

	Evaluation number	Efficiency improvement (%)	Peak position deviation (%)	Cavitation existence
NOMAD	20139	2.5	-1.6	No
	21988	2.4	-0.7	No
	22705	3.3	-2.1	No
	26587	1.8	0.1	Yes
EASY	27001	2.4	-1.5	No
	30989	2.1	-1.8	No
	37356	1.9	-0.9	Yes
	39686	2.3	-1.0	No

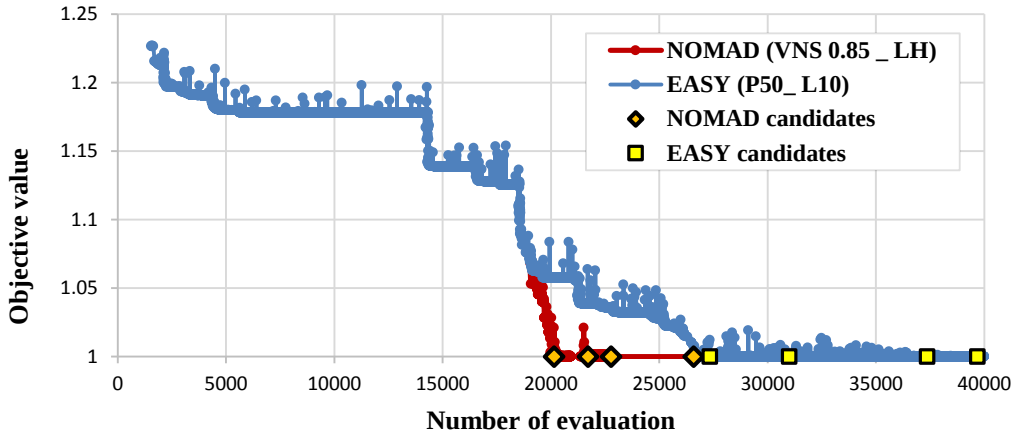


Figure 4-8 : Objective value improvement of feasible solutions obtained by the best NOMAD- and EASY optimizations

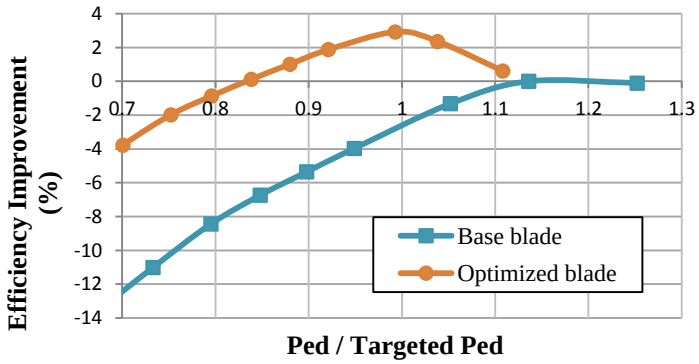


Figure 4-9 : Efficiency improvement of the optimized blade versus normalized power coefficient

The optimized blade is much shorter with a complex pattern of blade curvature. According to Figure 4-9, using length objective with three well-defined constraints has led to considerable efficiency enhancement (2.4%) at the right operating condition. In this figure, efficiency improvement has been calculated from the peak efficiency of the base geometry. Comparison of Figure 4-10 and Figure 4-11 shows the improvement of low-fidelity pressure curves along the blade to satisfy the pressure constraint on the targeted operating condition. Pressure coefficients have been normalized with the minimum and the maximum coefficients of the optimized blade. These pressure curves are relatively consistent with high-fidelity results. The proper arrangement of pressure curves along the optimized blade sections has led to an appropriate blade loading and negative load prevention. Satisfaction of tangential velocity constraint has significantly improved the tangential velocity profile at the runner outlet on the targeted operating condition, which plays an important role to minimize energy losses and to maximize the efficiency (see Figure 4-12).

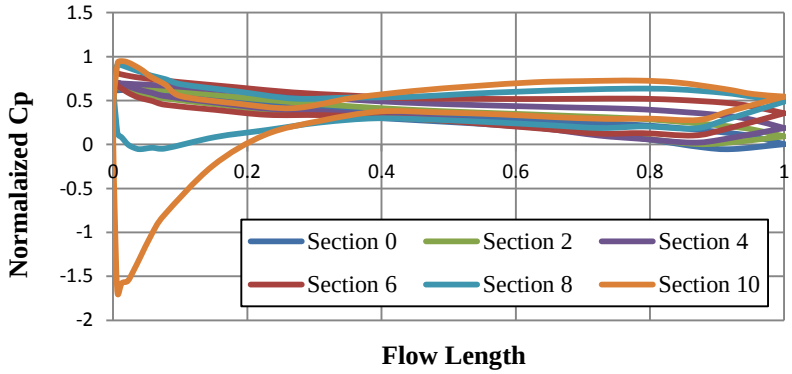


Figure 4-10 : Normalized pressure coefficient along the initial blade sections

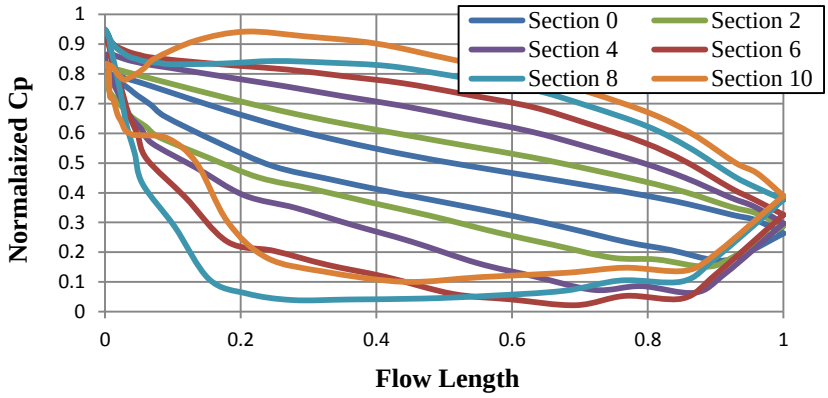


Figure 4-11 : Normalized pressure coefficient along the optimized blade sections

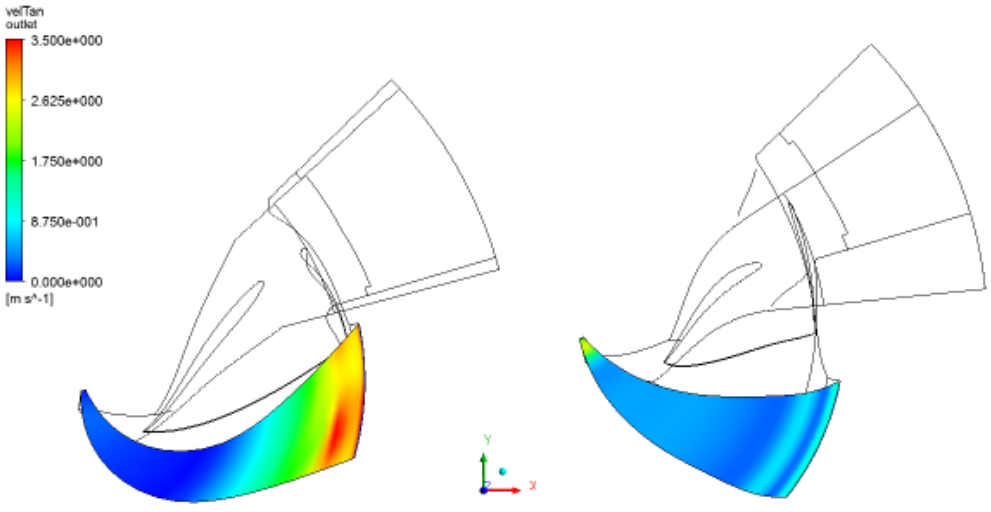


Figure 4-12 : Tangential velocity improvement at the runner outlet; left: base geometry, right: optimized geometry

The results indicate that no more design step is needed, since:

- All defined constraints have been satisfied in the first step.
- No relaxed constraint has been used in the optimization; so there is no need to increase the number of design variables to achieve feasibility of real constraints.
- High-fidelity evaluations have indicated that the main optimization target (i.e. right peak position) has been achieved properly, with a good efficiency. Therefore, no target correction is needed.

However, if there was a significant deviation from the targeted peak position, the operating condition used in the low-fidelity optimization constraints should be corrected in the next optimization step. We applied two linear corrections previously in optimizing a medium high-head Francis runner within three optimization overall loops [92]. Also, off-design operating points can be considered by adding new constraints and/or objectives dedicated to those operating conditions such as [117].

The optimized blade may be employed by designers as a good starting point to do very fine tunings in order to obtain other characteristics of a desired design. This final design step cannot be carried out within the optimization mainly due to the necessity of using a lot of geometrical parameters.

4.6 Conclusion

A robust multi-fidelity design optimization methodology has been developed to integrate advantages of high- and low-fidelity analyses, aiming to help designers to reach efficient turbine runners in reasonable computational time and cost. In the low-fidelity phase the automatic optimization loop is in charge of providing a lot of solutions using fast low-fidelity inviscid flow evaluations as a physic-based surrogate model. Considering important challenges of the design environment (such as non-linear non-convex design spaces), NOMAD and EASY have been chosen. NOMAD uses the MADS algorithm, which has demonstrated its power of local search in complex industrial applications. EASY employs an evolutionary algorithm, which is quite well-known for its global search capability.

Since the low-fidelity solver used in the optimization is not accurate enough and the efficiency is not directly reachable, a filtering unit selects a limited number of geometrically different candidates

which are dominant in their own neighborhoods. The computationally expensive high-fidelity phase is responsible for evaluating those candidates and choosing the best. These accurate results are valuable as well to calibrate low-fidelity optimization by the optimization correction unit.

The developed methodology demonstrated its advantages by designing a low-head Francis runner through a relatively low computational cost. Although the initial geometry was quite poor, all design targets were met in the first optimization step without any optimization tuning. The design targets were relatively simplified and only associated to the peak operating point. In the case of using more complicated targets with more design points, it is expected to use more optimization steps, corrected by high-fidelity results.

NOMAD and EASY optimization results indicate that each of them has its own abilities and drawbacks for the problem at hand, which come from the algorithms each of the optimizers employ. It may be concluded that for such a complex optimization problem, combining those two optimization algorithms can bring more optimization efficiency, while alleviates their disadvantages. While NOMAD, even assisted with LH, could not obtain the feasibility up to around 10,000 evaluations (25% of the maximum evaluation budget), EASY always achieves very big percentages of feasible solutions, without difficulty from about 1000 evaluations to the end of optimization. Therefore, it is a good candidate to apply for the first round of low-fidelity optimization, in order to have a wide global search of the design space within a couple of thousands evaluations. In the second round, NOMAD will gain from those promising feasible solutions obtained by EASY for deep local searches. NOMAD has proved its high performance to do the aforementioned task within a very few evaluations. By applying this new methodology, it is possible to cut significant computational cost and time, and achieve better optimized blade as a result of a better design space exploration.

4.7 Acknowledgement

The authors would like to thank Andritz Hydro Canada Inc. and NSERC, Natural Sciences and Engineering Research Council of Canada, for supporting this project. They also would like to gratefully acknowledge Nigel Murry, Christophe Devals, Maxime Gauthier and Evgenia Kontoleontos at Andritz Hydro Canada Inc. and Andritz Hydro GmbH for their contributions.

CHAPTER 5 ARTICLE 3: APPLICATION OF A TERRITORIAL-BASED FILTERING ALGORITHM IN TURBOMACHINERY BLADE DESIGN OPTIMIZATION

S. Bahrami[a], M. Khelghatibana[a], C. Tribes[a], S. Y. Lo[b], S. von Fellenberg[b], J. Y. Trépanier[a] and F. Guibault[c]

[a] *Mechanical Engineering Department, École Polytechnique de Montréal, Montréal, Quebec, Canada, H3T 1J4*
[b] *R&D Division, Andritz Hydro Canada Inc., 6100 TransCanada highway, Point-Claire, Quebec, Canada, H9R 1B9*
[c] *Computer Engineering Department, École Polytechnique de Montréal, Montréal, Quebec, Canada, H3T 1J4*

*Based on the paper submitted to the "Engineering Optimization".

5.1 Abstract

A territorial-based filtering algorithm (TBFA) is proposed as an integration tool in a multi-level design optimization methodology. The design evaluation burden is split between low- and high-cost levels in order to properly balance the cost and required accuracy in different design stages, based on the characteristics and requirements of the case at hand. TBFA is in charge of connecting those levels by selecting a given number of geometrically different promising solutions from the low-cost level to be evaluated in the high-cost level. Two different test case studies, a Francis runner and a transonic fan rotor have demonstrated the robustness and functionality of TBFA in real industrial optimization problems.

Keywords: filtering algorithm; multi-level design optimization; blade shape optimization

5.2 Introduction

Designing a turbomachinery blade currently relies extensively on the designer's intuition and experience. Although turbomachinery designers widely use computational fluid dynamic (CFD) tools, there is a growing need to integrate more tightly CFD analyses to obtain more automatic and efficient design optimization processes. Since expensive CFD evaluations and huge non-convex noisy design spaces are usually the biggest challenges, multi-level design optimization methods constitute one of the solutions.

In a multi-level optimization methodology, relatively low-cost analyses, as a substitute to highly accurate high-cost ones, evaluate objective function and constraint values. High-cost evaluations are performed outside the optimization to recalibrate the low-cost optimization problem and select the final design. Different types of low-cost methods have been employed for the low-level turbomachinery optimization, such as functional surrogate models (e.g. [67]), physics-based surrogates (e.g. [86]), coarser mesh (e.g. [72]), and reduced convergence tolerance (e.g. [71]).

In general, two classes of problems may be defined using multi-level optimization methods. In the first one, low- and high-level objectives are the same. The low-level optimization response should be corrected to satisfy consistency conditions with the high-level evaluations. Correction methods, such as response correction (e.g. [78]) and space mapping (e.g. [53]), try to calibrate input- or output corrections.

In the second class, low-level objectives are different from high-level ones, mostly because the high-level design objectives and/or constraints are not assessable by low-level evaluations, or they are too expensive. For instance, rotor efficiency maximization cannot be evaluated in a low-level optimization using an inviscid flow solver (as a physics-based surrogate), mainly due to the lack of physics involved to capture viscous phenomena. As an alternative, proper pressure distribution on the blade or blade boundary velocity profiles can be optimized in the low-level.

While previous works in the literature have mostly focused on the development and implementation of the first class of multi-level optimization, this article aims to address the second class. In this class, the optimized solution (in a single-objective optimization) or the set of Pareto solutions (in a multi-objective optimization) may not be the optimal one/ones from the high-level objective perspective. Thus, in order to increase the chance of reaching a good design, there is a need for a method to select a limited number of promising solutions to be evaluated by high-cost analyses, which are not necessarily dominant from the low-level optimization point of view.

In functional surrogate optimization, selection of solutions is based on certain rules called infill criteria, and the selected solutions are called infill points. Infill points are analyzed in order to validate the surrogate, as well as its accuracy enhancement. The exploitation infill method uses the surrogate minimum as the new infill point. For instance, it was employed in Approximation Model Management Optimization (AMMO) [73]. Also the Surrogate Management Framework (SMF) uses the exploitation method. SMF was developed and employed by Serafini [82] and Booker et

al. [38] to handle the use of surrogate methods in pattern search optimization techniques. The exploration infill method utilizes new infill points between the existing ones to build a globally accurate surrogate model (e.g. [60]). In global optimization, exploitation and exploration can be utilized together (e.g. research of Gutmann [119] and Jones [120]).

The main goal of the present work is to propose a newly developed filtering algorithm, which can be used as a key component of the second class of multi-level optimization problems. This method is similar to infill methods. This work also aims to apply the developed methodology on the multi-objective shape optimization of two turbomachinery blade design problems.

5.3 Territorial-Based Filtering Algorithm (TBFA)

The filtering process is in charge of selecting a given number of promising solutions among all feasible points obtained during the optimization, that are geometrically different and dominant in their own territories; hence we name 'territorial-based filtering algorithm'.

TBFA has been designed for multi-objective optimization. However, it is easily adapted for single-objective optimization as described in article of Bahrami et al. [113]. While the focus of that article was a low-head Francis runner test case without emphasis on the filtering process, the current article concentrates mostly on the TBFA and its steps. TBFA components are described in the following sub-sections.

5.3.1 Feasibility and Pareto front determination

The set of all N-dimensional feasible solutions (M solutions) obtained during optimization is written as:

$$Y^* = \{Y_1, Y_2, \dots, Y_m, \dots, Y_M\} \tag{5-1}$$

$$Y_m = \{y_{m,1}, y_{m,2}, \dots, y_{m,n}, \dots, y_{m,N}\} \tag{5-2}$$

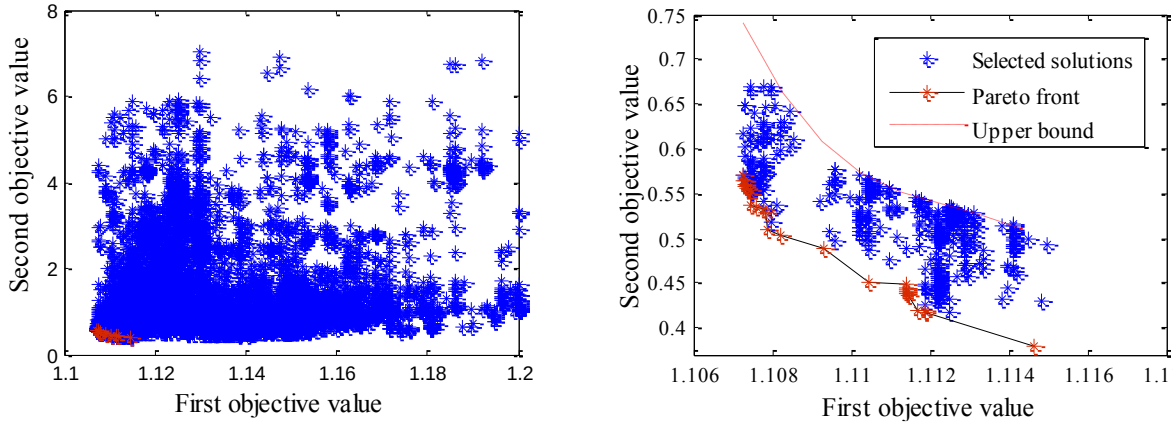
Here, we are considering the case where two objective functions, f_1 and f_2, are minimized, but the method can be easily generalized.

In the first TBFA step, feasible solutions are captured after completion of a first optimization step. Then, the Pareto front approximation is identified. If no feasible solution is obtained, constraints

may be relaxed and the optimization is re-launched, or those solutions that are close enough to feasibility are selected, based on thresholds provided by the designers.

5.3.2 Objective-based filtering

Promising solutions resulting from the optimization are supposed to be close to the Pareto front. Therefore, a band of feasible solutions is selected (i.e. $Y_B^* \subseteq Y^*$), which is limited to the Pareto front on one side of the objective space. The outer side of this band is a quadratic approximation of the Pareto front (see Figure 5-1 : Objective-based filtering process: (a) before (b) after filtering.). The width of this band (i.e. perpendicular distance from the Pareto front) should be adapted to the case at hand. It is mainly dictated by the distribution of feasible solutions in the objective space.



(a) (b)

Figure 5-1 : Objective-based filtering process: (a) before (b) after filtering.

5.3.3 Space mapping

The working space is scaled and mapped to recognize the amount of similarities/dissimilarities between solutions. The set of previously selected solutions, Y_B^*, is mapped into $Z_B^* = \{Z_1, Z_2, \dots, Z_b, \dots, Z_B\}$ using the following operator:

$$Z_{b,n} = \frac{G_n \times \left(y_{b,n} - \min_{b=1,\dots,B} y_{b,n}\right)}{\max_{b=1,\dots,B} y_{b,n} - \min_{b=1,\dots,B} y_{b,n}} \qquad (5\text{-}3)$$

Where the G_n is a set of positive integers given as inputs to control the granularity of the mapping process. The operator shrinks or stretches the scales of original axes in order to standardize the design space and provide a positive multidimensional Euclidean space. It also allows dividing the

design space into hypercubes of unit length sides. Each hypercube can be indexed by N positive integers each dedicated to one dimension. A similar technique was applied in the research of Postaire and Zhang [101] to prepare a set of observations for binary morphological transformations. Figure 5-2 shows an example of mapping and meshing applied on a two-dimensional design space, where $G_1 = G_2 = 6$.

The developed mapping provides population concentration information in boundary cells (i.e. hyper-cubes beside design space borders), which is helpful to illustrate whether the borders need to be changed. Later, in the first test case, the improvements achieved by bound modifications are described.

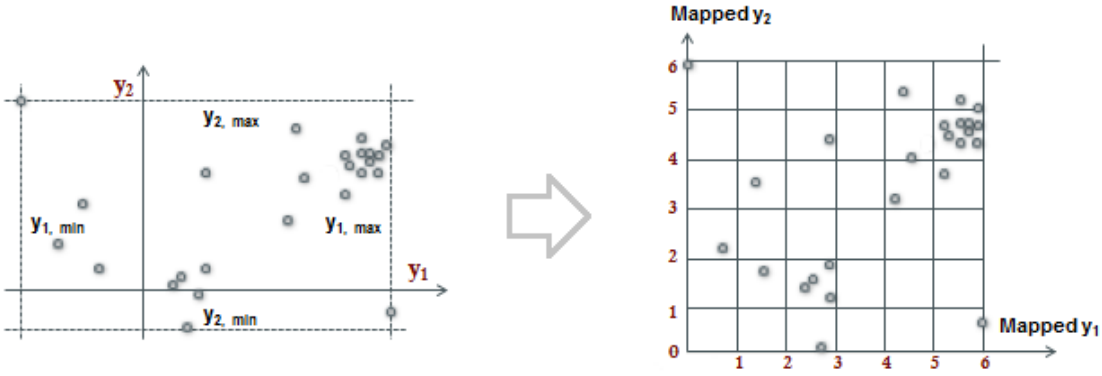


Figure 5-2 : Design space mapping process

5.3.4 Sieving

The aim of the sieving process is to select at most one solution in each hypercube based on a given granularity. Mutual-solution hypercubes contain more than one solution. In each of them the local dominant solution is selected. If there is more than one optimal solution, the local Pareto front approximation is determined inside the hypercube. Then, the territorial-Pareto solution closest to the hypercube center in the design space is chosen as the candidate of the hypercube. By this methodology, the most promising solutions are sieved, which are globally dissimilar enough. Figure 5-3 shows an example of the sieving process applied on a two-dimensional design space.

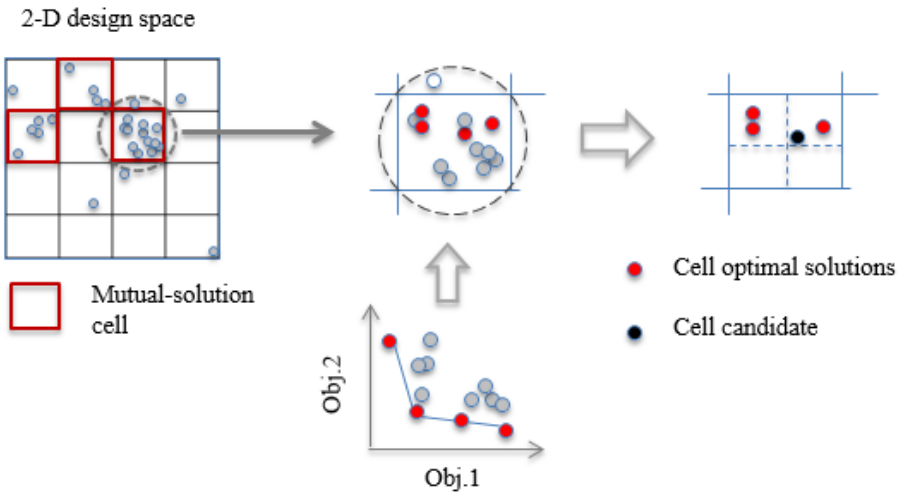


Figure 5-3 : Illustration of sieving process; selecting at most one candidate from each hypercube

5.3.5 Cluster formation

The sieving process usually selects a large number of solutions, and more filtering is needed. Therefore, a robust inexpensive clustering method is used. This new clustering is based on the core territory concept. A similar method was used by Dominique et al. [102]. In an iterative process, the method tries to choose a desired number of candidates, C, among previously sieved solutions ($Z_C^* \subseteq Z_S^*$):

Each selected solution is a core (i.e. center) of a single cluster territory delimited by a hypersphere. All hyperspheres have the same radius. The cores of hyperspheres are selected as the final design candidates. To choose the cores, the proposed algorithm needs an initial core and an initial radius. A good choice of initial core is a Pareto member close to the middle of the Pareto front approximation. The initial radius is set to the average distance of the first core to each border of the multi-dimensional design space. Figure 5-4 shows clustering of solutions for two values of 'C' in a two-dimensional design space with the same initial core, Z_1. The clustering algorithm is as follow:

Algorithm 1: Clustering algorithm

Given Z_S^* and C

Select Z_1 in Z_S^* (the closest to the Pareto front approximation center)

Initialize R (cluster radius, calculated from Z_1 distances from design space boundaries)

While $|Z_C^*| \neq C$

Begin

Initialize Z_C^* by Z_1

For each $Z_s \in Z_S^*$

If Z_s is outside Z_C^* territories

put Z_s in Z_C^*

Update Z_S^* to remove those points located inside Z_C^* territories

End

End

Update R: $R = R \times k\left(\frac{|Z_C^*|}{C}\right)$

$k > 1$ if $|Z_C^*| > C$ (expansion)

$0 < k < 1$ if $|Z_C^*| < C$ (shrinkage)

End

End

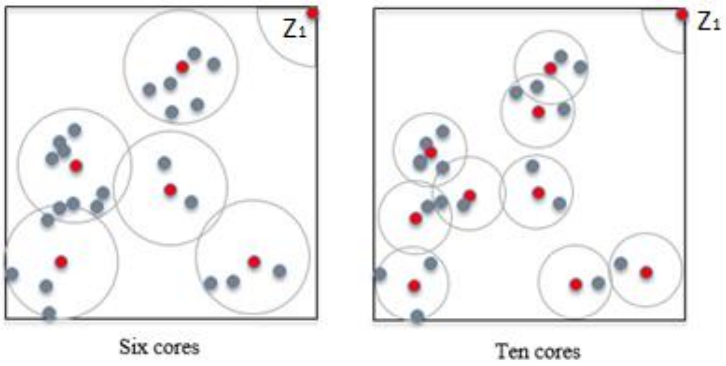


Figure 5-4 : Schematic of clustering in a 2-D design space

5.4 Test case 1- hydraulic turbine runner blades

A medium-head Francis turbine runner has been chosen to be optimized for a given set of operating conditions. This machine should have its best efficiency point (BEP) at a given operating condition defined by the power coefficient (P_{ed}) and speed coefficient (N_{ed}). Also, it has to work in the absence of cavitation at BEP, as well as at a full-load operating condition with 22% higher power

coefficient and about 2% higher speed coefficient. Figure 5-5 shows the optimization flowchart, which is described in the following subsections. The multi-level optimization methodology and its formulation have been presented in previous publications [92, 117].

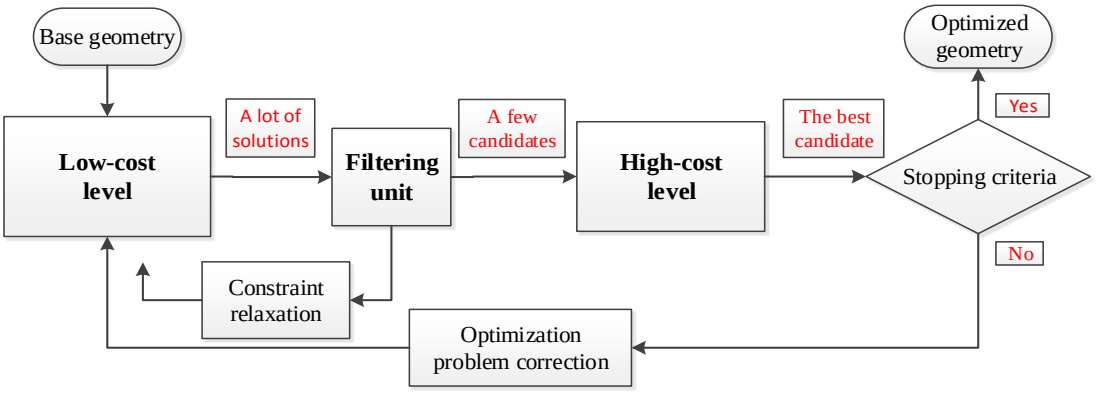


Figure 5-5 : Multi-level optimization flowchart

5.4.1 Low-cost level

This level contains an automatic low-fidelity optimization loop, which uses a physics-based surrogate optimization. Blade geometries are parameterized through an in-house software developed by Andritz Hydro Canada Inc. [109]. Table 5.1 shows independent parameters of the base geometry, and those employed as design variables.

Table 5.1 : Design variables and their bounds

Parameters	design variables	Bounds
Blade curvature	12	5 to 90 degree
Blade length	3	-50% to 15% (relative)
Blade leading edge	4	-12% to 12% (relative)
Band contour	4	-7% to 7% (relative)
Crown contour	Fixed	-
Blade thickness	Fixed	-
Number of blades	Fixed	-

The parameterization software creates the new geometry using design variable values received from the optimizer, and prepares it for CFD evaluations. A potential flow solver evaluates the flow domain.

NOMAD [93] handles the optimization problem using the Mesh Adaptive Direct Search algorithm (BIMADS [51]). For this case study, 80% of the optimization computational budget was dedicated to global search using the VNS method [50].

Two objective functions and three constraints have been defined. For the first time, a blade length objective was considered with an outlet velocity objective. The first objective aims to minimize the average length of the blade, L_s in Equation (5-4), calculated in 'S' blade sections, mainly to reduce blade friction losses. To obtain more flexibility, each blade section length can be weighted using k_s. The second objective tries to reach a flat-zero tangential velocity target at BEP (i.e. *OP1*) at the runner outlet illustrated in Figure 5-6. The optimizer minimizes the objective function formulated in Equation (5-5) as the difference between the tangential velocity target, V^*_{Tan}, and the one obtained, V_{Tan}, on a set of discrete radii, r_k. This objective aims to reduce the runner energy loss as well as improving the draft tube performance.

Equations (5-6) and (5-7) formulate two constraints that control the difference between the cavitation pressure, Pr_{Cav}, and the minimum pressure on the blade, Pr, to prevent cavitation at BEP and at the given full-load condition (i.e. OP2). In addition, one constraint has been defined to prevent negative blade loading (F_s in Equation 5-8) at BEP in order to prevent significant efficiency penalty in high-fidelity flow analysis. A maximum of 40,000 evaluations were performed for the low-cost level optimization, which ensures to obtain a good approximation of the Pareto front.

$$f_1^l(Y) = \sum_{s=1}^{S} k_s \, L_s(Y) \tag{5-4}$$

$$f_2^l(Y) = \sum_{k=1}^{K} abs\left[V(Y)_{Tan(r_k)} - V^*_{Tan(r_k)}\right]_{@\,OP1} \tag{5-5}$$

$$g_1^l(Y) = [Pr_{Cav} - \mathrm{Pr}(Y)]_{@OP1} \leq 0 \tag{5-6}$$

$$g_2^l(Y) = [Pr_{Cav} - \mathrm{Pr}(Y)]_{@OP2} \leq 0 \tag{5-7}$$

$$g_3^l(Y) = \left[\min_{s=1,\ldots,S} F_s(Y)\right]_{@OP1} \geq 0 \tag{5-8}$$

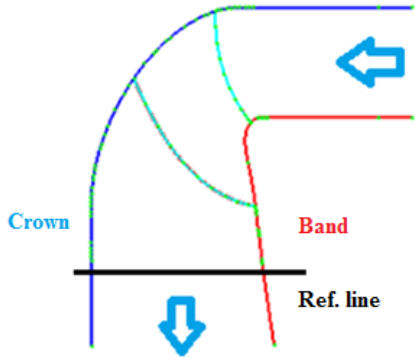
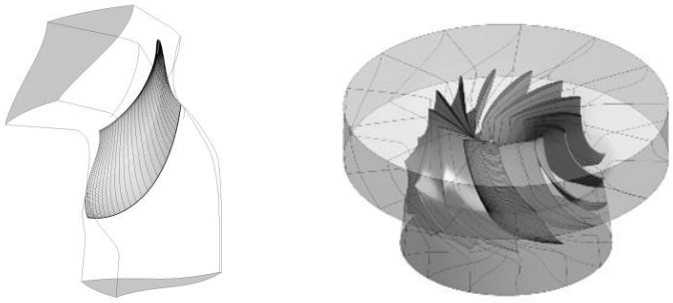


Figure 5-6 : Schematic of a runner flow passage in R-Z view, with the velocity reference line at the runner downstream

5.4.2 High-cost level

The high-cost level is in charge of accurate analyses of selected promising candidates in order to choose the best in terms of efficiency at the targeted operating condition, without cavitation. This phase consists of grid generation, viscous flow evaluations, and post processing. The accurate high-cost level data can also be used to calibrate the low-level optimization. An in-house automatic code of Andritz Hydro Canada generates an appropriate grid for each candidate geometry. The commercial Navier-Stokes code ANSYS-CFX has been employed for the flow field simulations using the standard two-equation k-ε Reynolds-Averaged Navier-Stokes (RANS) turbulence model. Details of the methodology, integrated tools and validations can be found in the references [105-107]. Post processing of the results provides runner efficiency and cavitation detection for each design candidate. Figure 5-7 shows an example of single-blade and the whole runner computational domain.



(a) (b)

Figure 5-7 : (a) computational flow domain containing meshed blade surfaces, inlet (top) and outlet (bottom), (b) runner flow computational domain with 13 blades

5.4.3 TBFA functionality and results

The TBFA connects the two levels explained previously. The results of the first filtering step indicated that 4744 feasible solutions have been obtained from the optimization, which was 12% of all evaluations. The study of the design space size pointed out a big feasible solution concentration near two design variable bounds; the lower bound of mid-section length (design variable 18 in Figure 5-8), and the lower bound of a leading edge point located on the crown. Therefore, the selected candidates of the filtering process have not been sent to the next level, and instead, the design space was corrected. The bounds of those design variables were shifted 25% downward. Consequently the size of design space remained constant.

The corrected design space was sent to the low-fidelity phase, and was employed in a new optimization. Figure 5-8 shows the big improvement of solution distribution from design variable 18 point of view. The corrected design space resulted in 6247 feasible solutions, which shows 32% increase. The best values of the first and the second objectives were also improved 6% and 85% respectively.

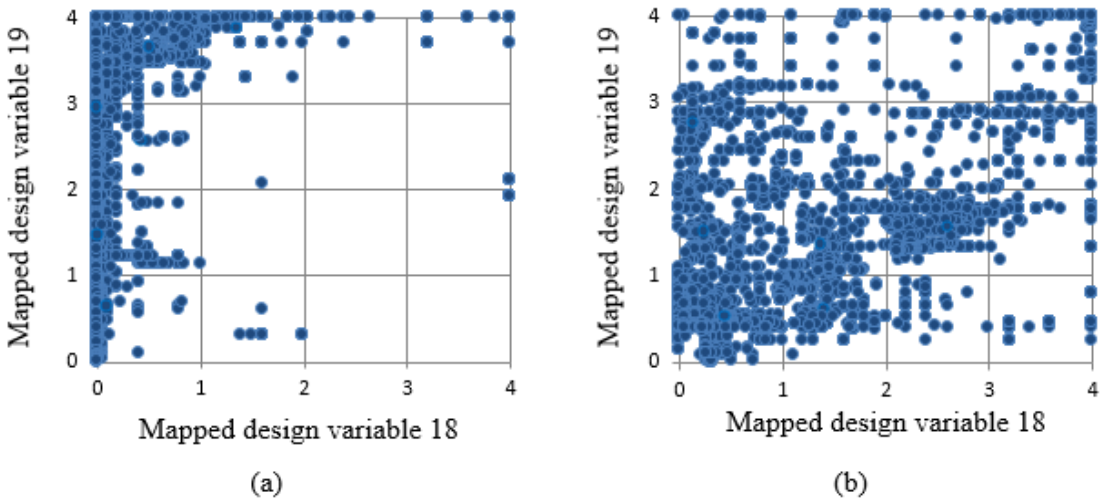


Figure 5-8 : Distribution of feasible solutions in the mapped design space, before (a) and after (b) applying the corrected design space

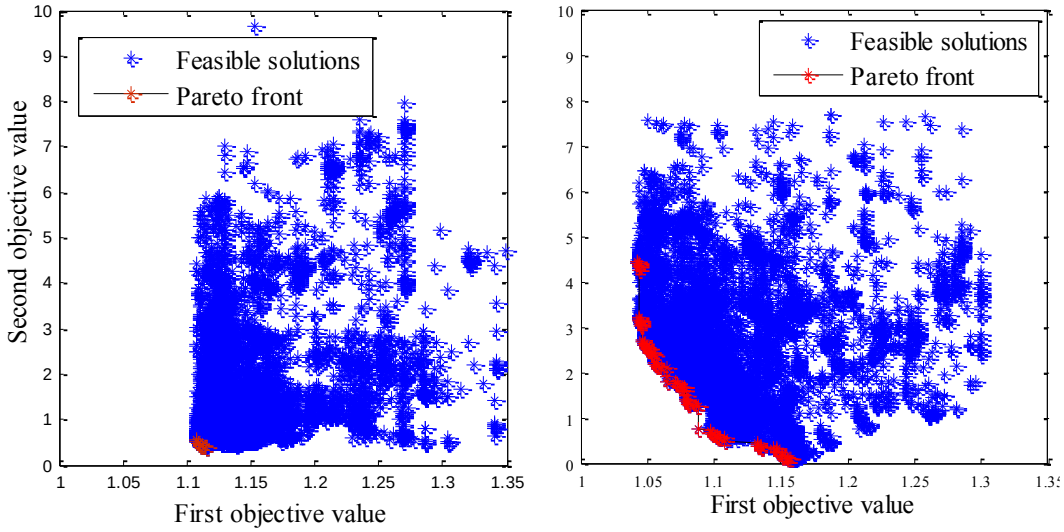


Figure 5-9 : Distribution of feasible solutions and Pareto front approximation in the objective space, before (a) and after (b) applying the corrected design space

Figure 5-9 shows Pareto front approximations and feasible solution distributions obtained by using the initial and the corrected design spaces. While the first Pareto front approximation is quite short containing 24 solutions, a much larger Pareto front approximation including 112 solutions has been achieved in the second optimization. Figure 5-10 shows the selected bound of feasible solutions close to the Pareto front obtained by using the corrected design space. Different granularity size can be chosen in the sieving process of TBFA. The smaller the filtering mesh size, the more sieved solutions are found. However, to obtain a given number of filtered candidates at the end, the clustering unit should compensate that and eliminate more solutions. Thus, it is preferable to keep the sieving mesh size large enough (i.e. smaller number of divisions). Figure 5-11 shows an example of using two different granularity sizes for a set of solutions. As it was expected, increasing the number of divisions increased the number of sieved solutions and sieved Pareto front members. It should be noted that by doubling the mesh size, the number of mapped hyper-cubes increases in the ten million orders of magnitude, in the problem at hand with a 23-dimensional design space. Therefore, the granularity size has been selected equal to four for both test cases.

Finally, the clustering unit selected six promising candidates transferred to the high-cost level. Figure 5-12 illustrates the location of those candidates in the objective space with their evaluation numbers. This figure also shows a reduced density of solutions inside the band and in the Pareto front due to sieving performance.

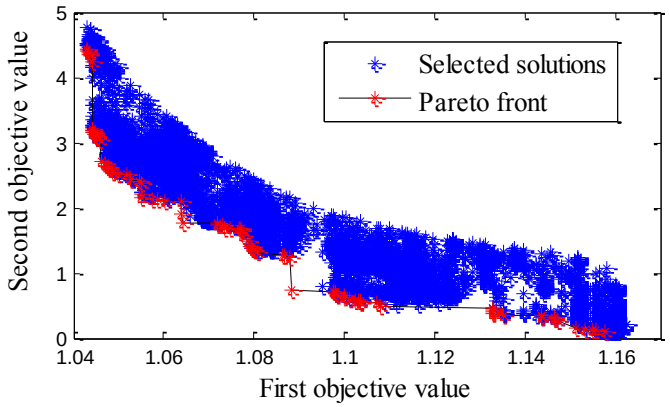


Figure 5-10 : Selected solutions before the sieving process

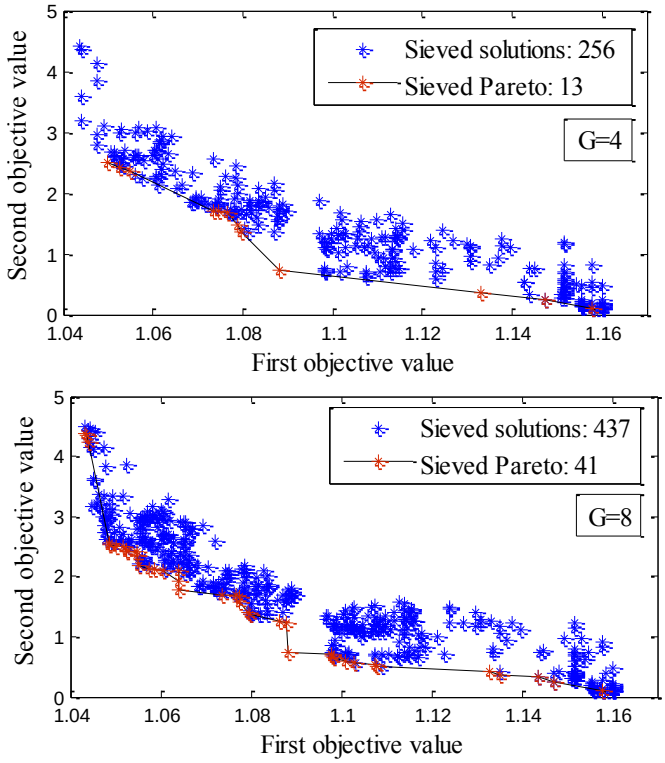


Figure 5-11 : Effect of different sieving grids

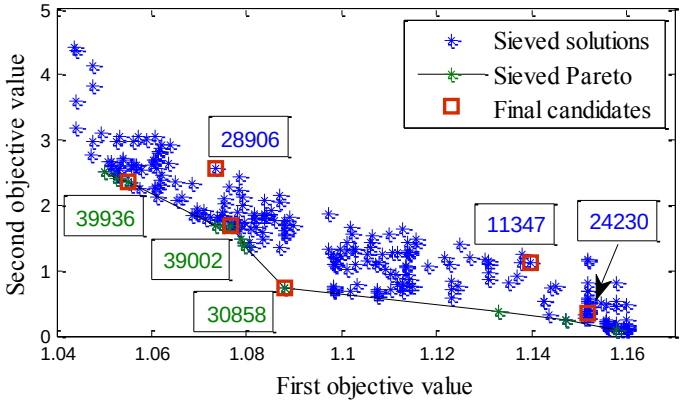


Figure 5-12 : Final candidates selected by clustering unit

Filtered candidates were investigated in the high-level. Post processing of Navier-Stokes analysis results determined the main design goals, which have been shown in Table 5.2. Efficiency improvement was measured at the targeted BEP. These relatively big efficiency enhancements are due to using a poor base geometry with a low efficiency. Table 5.2 shows that two of the three most efficient candidates do not belong to the Pareto front approximation. This point demonstrates the necessity of not focusing only on the low-cost level dominant solutions, and the relevance of the proposed filtering method.

Table 5.2 : Selected candidates and their performance improvements

	Pareto membership	Efficiency improvement (%)	Cavitation at BEP	Cavitation at full-load
39936	Yes	6.9	No	Yes
28906	No	6.6	No	Yes
39002	Yes	6.4	No	No
30858	Yes	7.0	No	Yes
11347	No	7.0	No	Yes
24230	No	7.4	No	Yes

Although none of the candidates cavitates at BEP, almost all of them cavitate at the full-load condition. This indicates that discrepancies in the pressure distributions obtained by potential- and Navier-Stokes flow solvers are intensified at the full-load condition, mainly due to stronger viscous phenomena at this operating condition. Therefore, the optimization problem needed to be corrected by revising the pressure constraint at the full-load condition. Thus, the best candidate, # 24230, was selected and sent back to the low-cost level to be employed as the new base geometry. By applying a linear correction, an increase of 0.04 unit was applied to the non-dimensional pressure

(height of water/net head) value used in the cavitation constraint. The optimization, filtering and viscous flow evaluations were repeated, and the best candidate was chosen after analyzing the high-cost level results. This candidate showed a 7.3% efficiency improvement in the absence of cavitation for both operating conditions. The results indicate that the correction of the full-load cavitation constraint caused elimination of cavitation from almost all selected candidates. Figure 5-13 and Figure 5-14 compare the pressure curves of candidate # 24230 and of the final candidate, obtained using the corrected optimization problem, using potential and viscous flow solvers respectively. To show the cavitation margin, for each candidate, only one pressure curve has been presented, which is dedicated to the blade section that has the minimum pressure. Figure 5-14 shows that the optimization correction was quite effective to remove the full-load cavitation in viscous flow results.

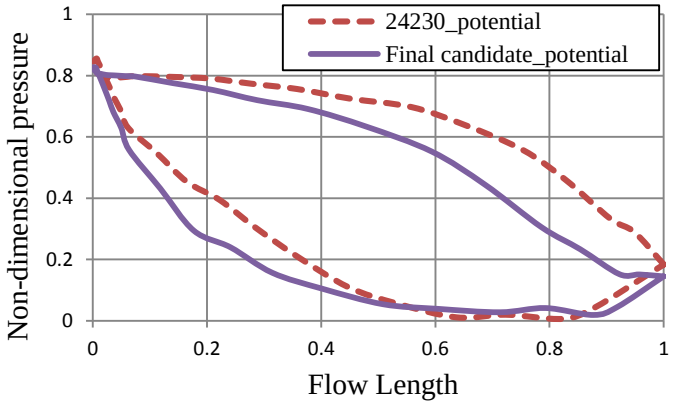


Figure 5-13 : Pressure distribution obtained by potential flow evaluations at OP2

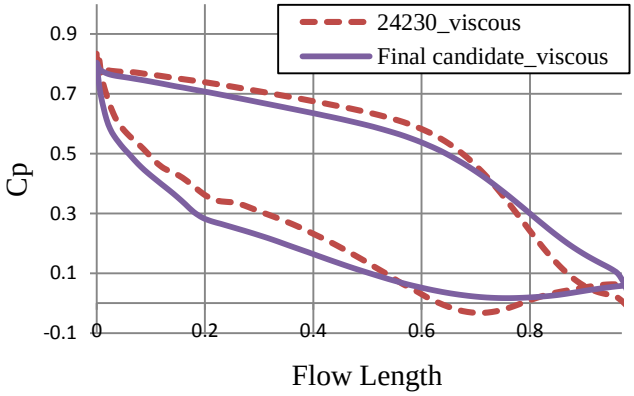


Figure 5-14 : Pressure distribution obtained by viscous flow evaluations at OP2

5.5 Test case 2- transonic fan blades

A Compact configuration of modern transonic compressors is achieved by maximizing pressure ratio per stage to reduce the overall size and weight.

The NASA rotor 67, a transonic axial-flow compressor rotor blade, has been used as the second test case study. This rotor has 22 blades and is the first stage of a two-stage fan designed in the late 1980's at the NASA Lewis Research Center. The NASA rotor 67 has become a well-known test case employed in different design optimization frameworks such as researches of Oyama et al. [121] and Pierret et al. [122]. Operational details near the peak efficiency and stall have been presented by Strazisar et al. [123].

5.5.1 Aerodynamic shape optimization algorithm

For this test case, the same flow solver with the same accuracy is employed in both levels. However, a computationally expensive high-cost level objective function is replaced with a less-expensive one in the low-cost level optimization loop. After the low-cost level optimization, the proposed TBFA selects a few promising candidates among feasible solutions. These candidates are then evaluated using the high-cost level in order to obtain the values of the primary objective functions. Post-processing of high-cost level results indicates the real objective values in order to select the best design, or a set of the best candidates to be transferred to the low-level as the new population. Also those computationally expensive results determine the correlation between the main objective and the substitute one.

5.5.2 Blade parameterization, grids and CFD evaluations

A three-dimensional fan blade parameterization method developed by Lupien [124] is used. This method provides a control on the blade geometry in the span-wise direction in order to prevent any inconsistency. In this study, 10 design variables including β angles and their derivatives are selected as design variables.

The flow simulation is performed in ANSYS CFX 14.5 using the Shear Stress Turbulence (SST) model. The computational domain which consists of a single blade passage in the rotational frame of reference is represented in Figure 5-15. All details of CFD simulations including the boundary conditions are available in a previous publication [125].

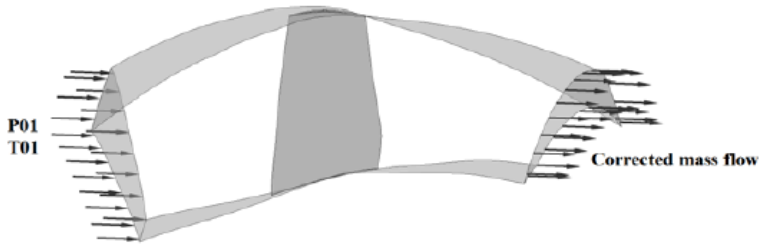


Figure 5-15 : Computational domain. Total pressure and total temperature at the inlet, corrected mass flow at the outlet boundaries [125]

The mesh used in this study has been created using the H/J/C/L-grid method including an O-grid type in ANSYS TurboGrid 14.5. A coarse grid with about 150,000 elements is used for evaluations in the course of the optimization process. A fine grid with about one million elements is employed to validate the final blades. The computational tool has been validated in previous publications [125, 126] by comparing the results obtained from the simulation with the experimental data from the work of Strazisar et al. [123].

5.5.3 Optimization features

This investigation aims to improve the NASA rotor 67 by considering two objectives. The first objective is to maximize the isentropic efficiency (i.e. minimizing the negative efficiency) at near-peak efficiency point at design speed (noted OP1). The isentropic efficiency is defined as:

$$f_1^l = -\eta_{is} = -\left[\frac{H_{O2,s}-H_{O1}}{H_{O2}-H_{O1}}\right]_{@OP1} \tag{5-9}$$

where $H_{O2,s}$ is the mass-averaged isentropic enthalpy at the outlet, H_{O1} is the mass-averaged total enthalpy at the inlet, and H_{O2} is the mass-averaged total enthalpy at the outlet. The second objective is to maximize the stall margin at near stall flow at design speed.

$$SM = \frac{PR_{stall}}{PR_{@OP1}} - 1 \tag{5-10}$$

The *stall* subscripts refers to the stall point, which is captured by incrementally decreasing the exit corrected mass flow boundary condition. The last numerically stable point is considered as the stall point. This iterative process, where several CFD evaluations are performed and convergence behavior of each must be analyzed, makes this calculation quite costly. Therefore, the stall margin objective function is substituted for isentropic efficiency at near stall condition, which requires

only one CFD evaluation. A previous study by Luo et al. [127] has reported on the potential of improving stall margin by improving near-stall efficiency. In the current study, further investigation is performed to verify whether stall margin has been improved by enhancing near-stall efficiency.

The fan is designed to produce a certain mass flow rate and pressure ratio for the engine cycle. To control the design mass flow rate, a corrected mass flow boundary condition is imposed at the outlet of the computational domain. The pressure ratio is controlled in the optimization problem by imposing a constraint as follows:

$$g^l = \left[abs\left(\frac{PR_{new} - PR_{R67}}{PR_{R67}}\right)_{@OP1} - 0.05 \right] \leq 0 \; ; \qquad PR = \frac{P_{O2}}{P_{O1}} \tag{5-11}$$

Where P_{O1} and P_{O2} are the mass-averaged stagnation pressures at inlet and outlet, respectively.

The Non-dominated Sorting Genetic Algorithm (NSGA-II) has been selected to handle the optimization problem. NSGA-II is a multi-objective Evolutionary Algorithm (EA) proposed by Deb [128]. In addition to the standard operators of any EA, an elite-preserving operator is imbedded in NSGA-II. Elitism ensures that the best solutions will not be lost and remain within the population on the subsequent generations [129]. For the problem at hand, a population size of 16 members for 30 generations has been employed (i.e. 480 blade evaluations).

5.5.4 TBFA functionality and results

The filtering process shows that (contrary to the first test case) feasible solutions have been distributed relatively well in the design space. Figure 5-16 shows all feasible solutions. A large Pareto front approximation with 37 members was achieved in this optimization, while feasible solutions concentrated close to it. The sieving process selected 21 geometrically different Pareto members and 88 feasible solutions. The next TBFA step, the clustering, was set to obtain 20 final candidates. Figure 5-17 shows the final candidates among sieved Pareto and sieved solutions. This figure indicates that only four of those candidates belong to the Pareto front approximation.

In Figure 5-18, the linear regression between stall margin and efficiency at OP2 obtained for Pareto points has a $R^2 = 0.7847$, while it is equal to 0.8541 for those candidates obtained by TBFA, which shows a tighter fit. More importantly, the correlation dedicated to TBFA candidates has been obtained from a larger variety of solutions in the design space (see Figure 5-19), so it is applicable

for a larger design space with good accuracy. In Figure 5-19, while Pareto members are usually geometrically similar and concentrated in a specific part of the design space, the final candidates selected by TBFA are well-distributed and geometrically as different as possible.

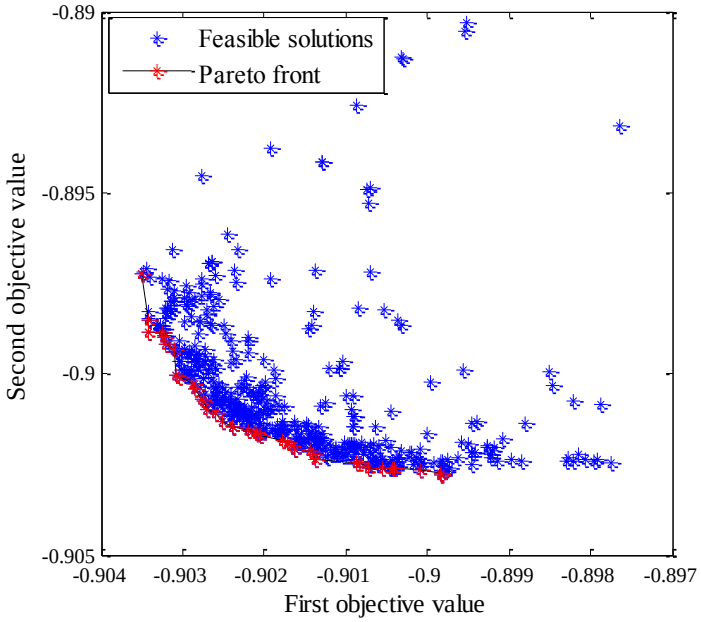


Figure 5-16 : Feasible solutions and Pareto front approximation in the objective space

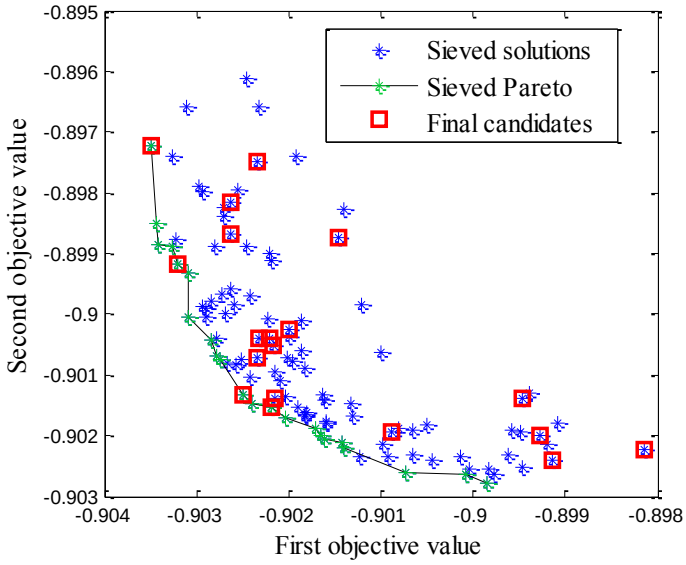


Figure 5-17 : Final candidates selected among sieved solutions

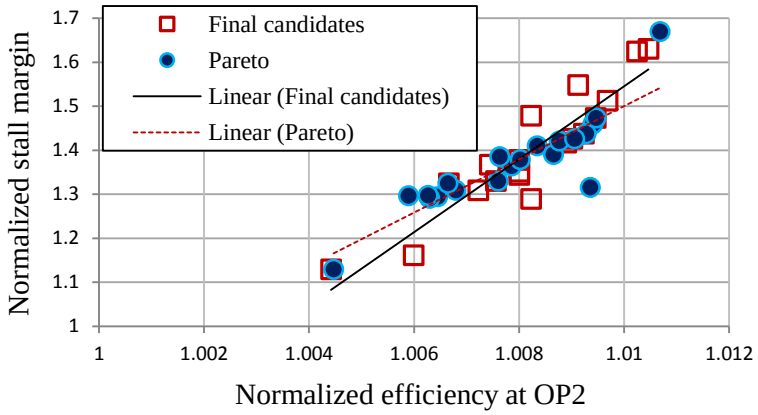


Figure 5-18 : Correlation of stall margin and efficiency of OP2

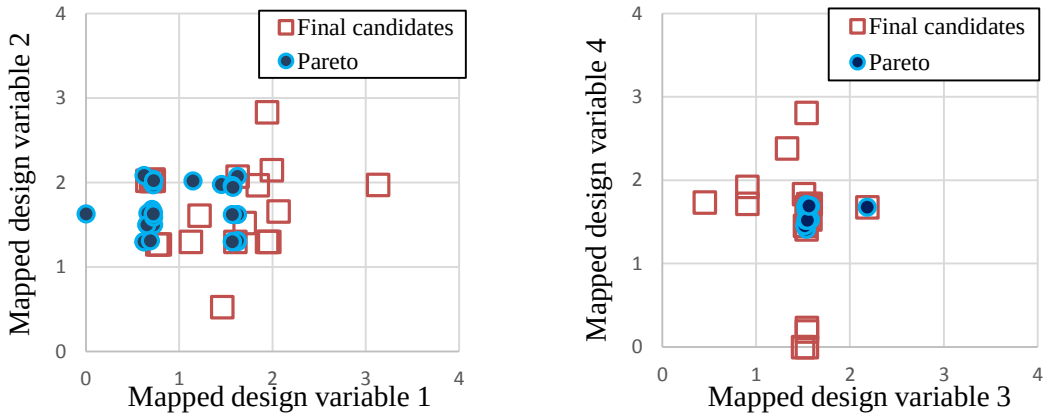


Figure 5-19 : Distribution of Pareto members and final candidates in the mapped design space

The final optimized blade was chosen based on the results of in-depth aerodynamic analyses using a fine mesh. The profiles of the optimized and the reference blade at hub, mid and tip sections are compared in Figure 5-20. Table 5.3 compares the performance parameters of this selected optimized blade with the NASA rotor 67. An efficiency increase of 0.62 points was achieved at OP1, while pressure ratio and mass flow rate at this operating point were, respectively, within 0.2% and 0.06% of the reference blade. At OP2, the efficiency was increased by 0.84 point. Moreover, the stall margin was improved from 2.1% to 3.6%.

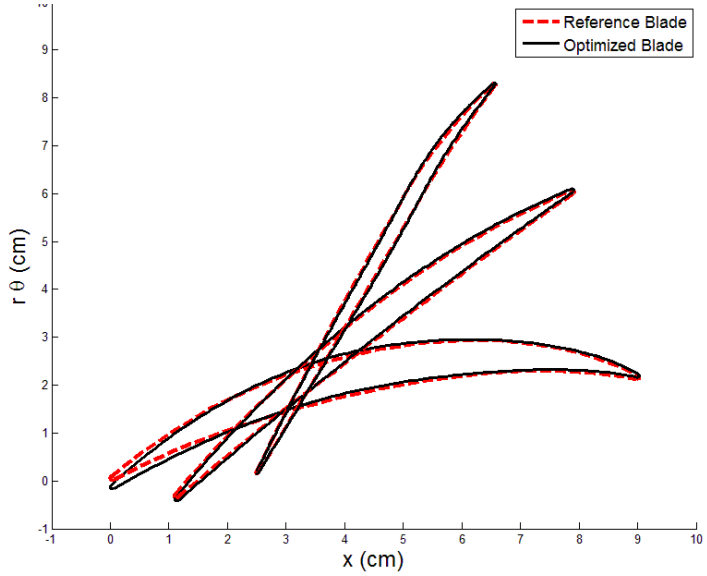


Figure 5-20 : Comparison of optimized and reference geometries

Table 5.3 : Performance comparison of blades

	Optimized Blade	**Reference Blade**	**Change**
$\boldsymbol{\eta_{is}}$ ***at OP1*** (%)	91.55	90.93	0.62
$\boldsymbol{\eta_{is}}$ ***at OP2*** (%)	90.69	89.85	0.84
SM (%)	3.6	2.1	+1.5
PR at OP1	1.641	1.645	-0.20%
$\boldsymbol{\dot{m}}$ ***at OP1*** (kg/s)	33.49	33.51	-0.06

5.5.5 Conclusion

A territorial-based filtering algorithm (TBFA) has been developed to connect different optimization levels within a multi-cost level design optimization methodology that uses substituted objectives in the low-cost level. It is in charge of selecting a certain number of promising candidates out of low-cost level optimization solutions, in order to be evaluated in the high-cost level. TBFA consists of five steps, which makes it robust, computationally inexpensive, and easy to implement. The proposed TBFA with multi-level optimization methodology demonstrated its capabilities when it was applied to two turbomachinery design cases.

In the first optimization problem, the shape of a medium-head Francis turbine runner blade was optimized. After the first set of optimization, TBFA preliminary step indicated that certain bounds of the design variables needed to be shifted. The new results showed significant improvements of objectives using the corrected bounds. This constitutes a practical application for design optimization as TBFA makes it easier to visualize distribution of candidates in the design space and choose appropriate design variable bounds. TBFA selected a few promising candidates for the high-cost level evaluations. While the high level results indicated cavitation constraint violations, the optimization problem was revised for the second time, and consequently the third set of optimization was launched using a corrected pressure constraint. The selected candidates evaluated in the high level met the design criteria. Within this hydraulic runner case study, TBFA proved its ability to select geometrically different design candidates that were dominant on their own territories, especially when it could capture good designs which were not located on the low-level optimization Pareto front approximation.

TBFA also demonstrated its functionality in the second case study, a bi-objective shape optimization of a transonic axial fan blade. While the same Navier-Stokes solver was used in both levels, one of the design objectives was substituted by a low-cost level optimization objective, since the original one was computationally expensive. Most candidates selected by TBFA did not belong to the Pareto front approximation. Also, high-cost level results indicated a good correlation between the original and the substituted objectives. This correlation is valid for a large variety of blades since it has been achieved using geometrically different blades obtained by TBFA.

5.6 Acknowledgment

The authors would like to thank Andritz Hydro Ltd. supporting this project. This work also was supported by the NSERC, Natural Sciences and Engineering Research Council of Canada, under Grant RDCPJ 441752-12.

CHAPTER 6 GENERAL DISCUSSION

A new The proposed Multi-Fidelity Design Optimization (MFDO) is a physics-based surrogate approach that brings more automation in the runner design process and reduces designer interactions. The main inspiration has come from the current runner design process and the developed methodology employs the same elements. Runner designers perform a large number of CFD investigations using fast and computationally cheap inviscid flow solvers to obtain the main features of good designs. They spend a big portion of their design time on manipulating one or different geometric parameters simultaneously by hand and evaluate new geometries usually using a potential flow solver. The biggest impact of the developed MFDO method is in this part, which tries to formulate the design criteria based on designers' experiences. Then an automatic low-fidelity optimization loop looks for good design candidates among different combinations of geometric parameters. Although this loop is still dependent on the designers' intuition, their interactions have been significantly reduced. Beside the gained efficiency, the optimizer can explore more systematically the design space than a designer can do, and can reach out of the designer's comfort zone. Thus, as it was seen several times in different test case studies during this Ph.D. project, the optimizer can achieve some good design candidates that are geometrically totally different from what was expected.

By using a physics-based surrogate such as a potential flow solver in the low-fidelity phase, high-fidelity accurate information has to be used to tune the low-fidelity optimization. Unlike functional surrogates, the physics-based surrogate cannot be modified for accuracy enhancement. However, high-fidelity information can tune the low-fidelity optimization formulation by correcting some optimization features such as objective functions and constraints. As it was demonstrated in the three articles, different types of corrections may be applied to the problem at hand. In the presented articles, two validated approaches were used to evaluate the new designs at the targeted operating conditions. In the two first articles, the high-fidelity CFD evaluation chain of Andritz Hydro Canada Inc. delivered a full efficiency curve for the given runner geometry. Therefore, it was possible to determine the error of the targeted peak operating condition with the peak obtained in that curve. Consequently, correction of peak positions was considered. In the third article another high-fidelity CFD approach was employed that do not produce an efficiency curve, but evaluated the geometry at the targeted operating condition by adapting internally the boundary conditions. Therefore correcting the targeted operating condition was not required. Instead, based on the

results, correcting the cavitation constraint was necessary, which was done by tuning the lowest allowed pressure in the targeted operating conditions.

Three different Francis runners were employed in case studies, in order to tackle different design challenges. In addition, having different optimization starting points led to demonstrate the flexibility of the proposed MFDO methodology. While the first initial point constituted an existing relatively efficient medium-head runner supposed to be redesigned for two given operating conditions, the second one was an inefficient low-head machine supposed to be designed for a targeted best efficiency point. However, the sequence of articles indicates the major evolution of problem formulation and optimization features. In the first article, desired tangential and meridional velocity profiles at the runner outlet were formulated as objective functions, and three constraints were used to control blade loading. The lowest parameterization fidelity was used in this article, which was enough to reach adequate number of feasible solutions and objective improvement. Our investigations indicated that the meridional velocity profile does not have such a high priority, since other design characteristics can control it. Therefore, it was removed from problem formulation for the next studies. Also, it was concluded that it is usually acceptable to obtain the tangential velocity profile within a certain bound. Consequently it was reformulated to move from objective to constraint in the second test case.

From the second test case, a new objective function, the blade average length, was formulated and implemented, which mainly aimed to minimize runner friction losses and consequently enhance the efficiency. It had a big effect on blade loading as well. To minimize the average blade length, the parameterized geometry required more flexibility. Therefore, two length design variables were added and allowed to vary independently. Also, the leading edge and band-side contour were taken into account by using five design variables. Three constraints associated with blade load slopes and smoothness were replaced with a new one that prevented obtaining a negative blade loading. Unlike the other two test cases, only one cavitation constraint dedicated to the peak operating condition was used in the second test case. However, obtaining the feasibility was the biggest issue; since it was reduced from 30% of solutions in the first test case down to 3% in the second one. That is why some new strategies were employed to obtain more feasible solutions with a better design space exploration. In the first strategy, different initial designs were employed in NOMAD optimizations, since the effect of initial design was intensified using such difficult constraints. The Latin hypercube method was utilized to help NOMAD initialization. In the second strategy, different

budget was dedicated to global search via the VNS method. However, the main improvement was achieved when an evolutionary algorithm was implemented. Although the evolutionary algorithm needed some adjustments, after that, it could reach a large number of feasible solutions (up to 64%). While it demonstrated powerful global search ability, its weakness in local search was obvious.

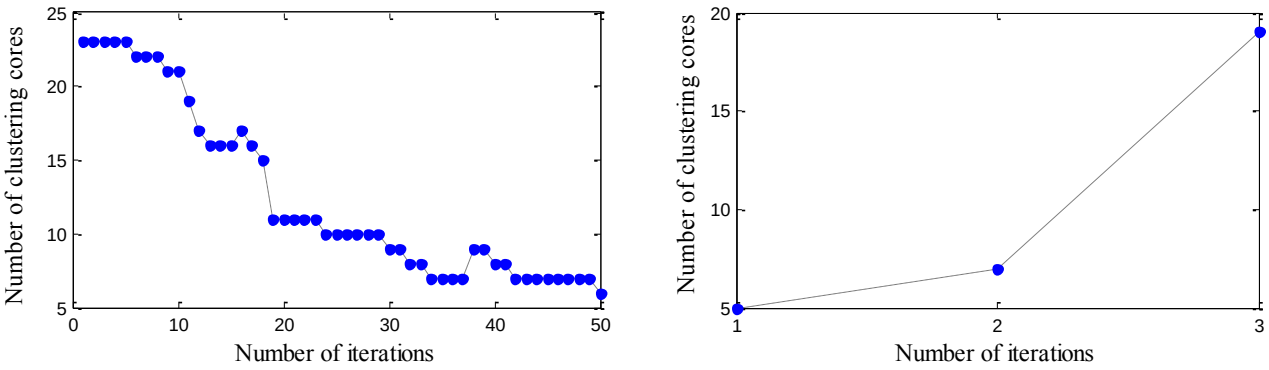
In the third runner case study, two more design variables were added to the blade leading edge, since it demonstrated its important role in the second case study. Therefore, the highest fidelity model with 23 design variables represented the runner blade in the third article. In this article the tangential velocity target was back as an objective. Another lowest pressure constraint was also added, associated with cavitation at a given full-load operating condition. Unlike cavitation at peak condition, the full-load cavitation could not be well-predicted and eliminated. Therefore, the low-fidelity constraint dedicated to it was corrected.

The newly developed Territorial-Based Filtering Algorithm (TBFA) is a key element of the proposed MFDO methodology. The TBFA has been built on the fact that low-fidelity objective functions are different from the high-fidelity design goals. For instance, in runner optimization using a potential flow solver, runner efficiency improvement or cavitation absence, as the main design goals, cannot be evaluated. As substitutes, other characteristics can be formulated to address these design goals within the low-fidelity optimization, such as desired velocity profiles at the runner outlet and low pressure limits. A substitute optimization approaches the real goals with some errors, not only due to employing different objectives and constraints, but also because of using a low-fidelity CFD solver. Therefore, the dominant optimization solutions may not keep their superiority in high-fidelity evaluations. The TBFA has been developed to fill this gap between low- and high-fidelity phases by selecting a given number of promising optimization solutions that are dominant on their own territories, while being geometrically different. The presented case studies demonstrated that the TBFA has successfully accomplished this task. The selected designs usually did not belong to Pareto front approximations. Also the TBFA explored all feasible solutions to choose different representatives from all over the design space. For instance, in the second article, a single length objective was used in the optimization, and all selected candidates by TBFA had the same objective value. However, they were relatively well-distributed in the design space, which caused obtaining divergent high-fidelity performances presented in

Table 4.3. The TBFA was investigated particularly in the third article with two case studies. In the Francis turbine case, the TBFA showed how it can improve the feasibility and objective values by

visualizing the solution concentration near the design space borders, and consequently modifying the design variable bounds. In the transonic fan case, the TBFA implementation caused selecting geometrically different blade geometries led to achieving a correlation between the stall margin and the efficiency, which is valid for a large variety of geometries.

The TBFA demonstrated its versatility within a lot of different applications where changing or eliminating some input points did not affect TBFA performance and stable functionality. The TBFA clustering, as the most computationally demanding element, has been quite fast and stable. In case studies, the clustering algorithm converged in a few iterations, based on the initial radii and cores; but it was always fast. For instance, Figure 6-1 shows the clustering convergence for hydro and aero cases presented in the third article. In the hydro test case, six candidates were selected among 269 sieved points after 50 iterations, which took several seconds. However, three iterations were enough to select 19 candidates in the aero test case. The TBFA convergence can be improved for other problems where the number of input solutions is much higher. For instance, the current linear interpolation used for hyper-sphere radius modification can be replaced with a higher order one.

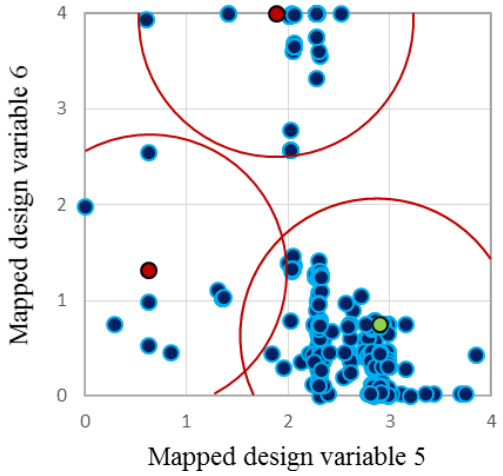


(a) Francis runner test case (b) fan test case

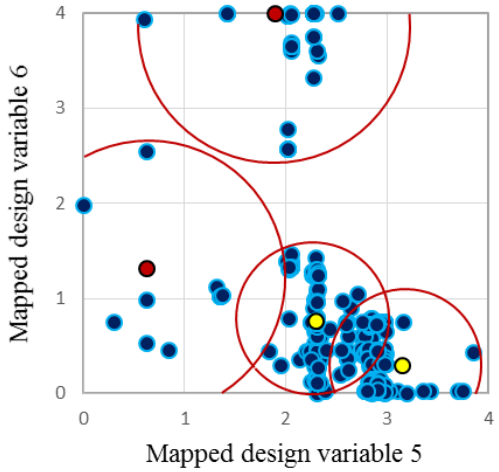
Figure 6-1 : Clustering convergence

The radius determination strategy can be improved as well. In the current clustering, all hyper-spheres have the same radius. Thus, only one design candidate represents other hyper-sphere members, regardless of the hyper-sphere population. In a new strategy, the size of hyper-sphere can vary to capture candidates more monotonically in terms of the population. For instance, in Figure 6-2 the hyper-sphere occupied with a large number of points is divided into two smaller hyper-spheres in order to have two representatives for this large population. This demonstration

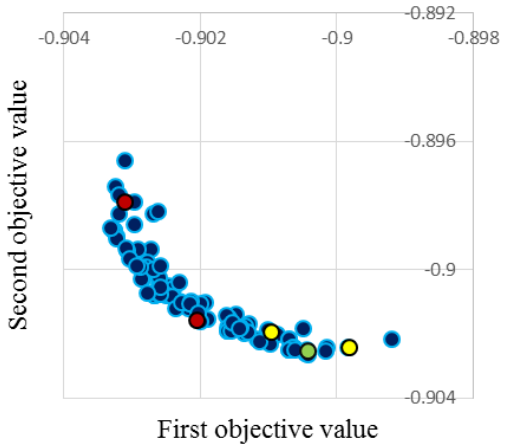
has been made from a projection of selected solutions obtained from the fan blade optimization. Part (c) of this figure indicates that the new strategy can lead to obtaining a larger variety of blade performances as well.



(a) Fixed radius hyper-spheres



(b) Variant radius hyper-spheres



(c) Selected candidates in objective space

Figure 6-2 : Demonstration of clustering with fixed and variant radii

CHAPTER 7 CONCLUSION AND RECOMMENDATIONS FOR FUTURE WORK

7.1 Conclusion and contributions of the work

A new multi-fidelity optimization methodology developed for hydraulic blade optimization problems is the main contribution that I made during my doctorate research program. Unlike other optimization methods and tools developed in academic settings, the involvement of Andritz Hydro Canada in this project led to the adaptation of the proposed methodology for the industry. In contrast to current design processes based on non-automated techniques with significant human interaction factor, the proposed design methodology can systematically explore unconventional configurations beyond historical comfort zones. The newly developed optimization methodology splits the computational burden between the low- and high-fidelity phases. The low-fidelity optimization approaches the high-fidelity goals indirectly through substituted objective functions and constraints. The high-fidelity phase accurately evaluates promising design candidates and tunes the low-fidelity optimization formulation (if required). The other contribution of this doctorate work is a new infill method, which connects two aforementioned phases via selecting promising design candidates using a Territorial-Based Filtering Algorithm (so-called TBFA). The TBFA is computationally cheap and easy to implement. It has also demonstrated its flexibility and stability in different test cases.

To determine the new methodology capability, I tackled runner optimization problems with multi-objectives and some difficult constraints, using significantly fewer computational resources and designer interactions. For instant, in a typical medium-head runner optimization (e.g. the one presented in the third article), about 100,000 potential flow evaluations are required, which takes about 40 hours without parallel optimization. Also, a maximum of 30 high-fidelity CFD analyses proved adequate. The corresponding computation time of these analyses strongly depends on the runner evaluation method and the computational resources. It takes about 15 hours using the same runner evaluation method employed in the third article and Andritz Hydro Canada’s cluster reported in the first article. Consequently the overall computational time of a design optimization process can be less than three days, which is quite acceptable. In this stage, the optimized runner usually poses to be good enough for the next design step, i.e. the detailed tuning. Proper

optimization configurations were also investigated including different objective functions, constraints, and design variables within the aforementioned runner optimization problems.

7.2 Recommendations for future works

Regarding the proposed multi-fidelity methodology and its implementation in hydraulic runner optimizations, the following subjects are suggested for future works;

1. Up to now, the proposed TBFA has demonstrated its capability within the aforementioned test cases. However, there is definitely room for improvement in different TBFA steps. The most effective modification can be applied to the last step of TBFA, the clustering. As it was mentioned in the last chapter, in each iteration of the clustering algorithm the same hyper-sphere radius is utilized all over the design space. However, using multi-radius hyper-spheres can scan more intelligently different scales that exist in the design space. Employing different sizes of hyper-spheres will lead to selection of more promising candidates from densely populated regions.
2. As it was demonstrated in the second article, the low-fidelity optimization can improve significantly by integration of local and global search algorithms. It can be vital in some shape optimization problems where the constraints are difficult to satisfy and the objective spaces are too complex. NOMAD still can be used since it allows the integration of a user search. In the Search step, NOMAD can employ a population-based algorithm such as EAs
3. Most modern turbomachinery optimization problems have more than one discipline. Even in a hydro blade optimization, there is a need to integrate other discipline considerations into the problem formulation, such as stress analyses and manufacturing limits. Therefore, in the next step, Multidisciplinary Design Optimization (MDO) methods can be taken into account to be integrated in the proposed multi-fidelity optimization methodology. The MDO integration can open a big window to aerospace optimization problems as well.
4. The number of design variables and the design space size constitute very important factors in derivative-free optimizations. Employing larger numbers of design variables gives more flexibility to designers to consider other important parameters, geometric or non-geometric, to optimize more features at the same time. However, its time penalty is considerable and sometimes makes the problem unsolvable. Model reduction methods can be quite helpful

in this regard. These methods can play a big role in large scale computational optimizations using costly CFD analyses. Model reduction methods usually decrease the design space dimensions by reducing the number of design variables. Consequently, less computational effort is required to achieve an optimum, which leads to considerable time gains. A future work can be the integration of these methods. In the first step, different model reduction methods should be studied to select the most appropriate one to employ in the developed methodology. Factor analysis, discriminant analysis, multidimensional scaling, projection pursuit, and Fourier model reduction are some of them. To implement model reduction methods, parametric dependence studies should be performed for the problem at hand.

5. In all hydraulic runner case studies presented in this thesis, a potential flow solver was used in the low-fidelity optimization loop. This solver prediction is acceptable for specific parameters in certain runner operating conditions. However, the error exponentially increases when the viscous phenomena are more severe and dominant in the flow field. Around the runner, it happens in part-load or full load operating conditions that are far from the best efficiency point. Also in other components, such as the draft tube, viscous effects play key roles. It is recommended to replace the potential flow solver with an Euler flow solver, which has more accuracy with fewer assumptions. Switching to Euler flow solver can even happen within the optimization process. Euler flow solver performance should be investigated in the aforementioned situations to determine how much global loop iteration will be reduced by added accuracy, and how much CPU time will be added. This trade-off study will lead to proper solver choices in different situations. In addition, such a study will enable designers to apply the proposed multi-fidelity methodology to design other components for more complex flow regimes.

BIBLIOGRAPHY

1. *10 Years of renewable energy progress*. 2015, Ren21: Renewable Energy Policy Network for the 21st Century: France.
2. *BP Statistical Review of World Energy*. 2015: England.
3. *Binge and purge*, in *The Economist*. 2009-01-22.
4. *2015 Hydropower status report*. 2015, IHA: International Hydropower Association.
5. Franc, J.-P.D., F. Avellan, and J.-L. Kueny, *La cavitation : mécanismes physiques et aspects industriels*. 1995: Presses universitaires de Grenoble PUG.
6. Zobeiri, A., *Investigations of time dependent flow phenomena in a turbine and a pump-turbine of Francis type - rotor-stator interactions and precessing vortex rope*. 2009, EPFL.
7. Wang, F.-j., et al., *Experimental Investigation of Characteristic Frequency in Unsteady Hydraulic Behaviour of a Large Hydraulic Turbine.* Journal of Hydrodynamics, Ser. B, 2009. **21**(1): p. 12-19.
8. Susan-Resiga, R. and S. Muntean, *Decelerated Swirling Flow Control in the Discharge Cone of Francis Turbines*, in *The 4 th International Symposium on Fluid Machinery and Fluid Engineering*. 2008: Beijing, China.
9. Ciocan, G.D., F. Avellan, and E.L. Berca, *Wall Friction Measurements: Application in a Francis Turbine Cone.* ASME Conference Proceedings, 2002. **2002**(36150): p. 317-322.
10. Tridon, S., et al., *Experimental analysis of the swirling flow in a Francis turbine draft tube: Focus on radial velocity component determination.* European Journal of Mechanics, B/Fluids, 2010. **29**(4): p. 321-335.
11. Avellan, F. *Flow Investigation in a Francis Draft Tube: the Flindt Project*. in *20th IAHR Symposium on Hydraulic Machinery and Systems*. 2000. Charlotte, North Carolina, USA.

12. Wang, S. and Z. Rusak, *The dynamics of a swirling flow in a pipe and transition to axisymmetric vortex breakdown.* Journal of Fluid Mechanics, 1997. **340**: p. 177-223.
13. Leclaire, B. and D. Sipp, *A sensitivity study of vortex breakdown onset to upstream boundary conditions.* Journal of Fluid Mechanics, 2010. **645**: p. 81-119.
14. Susan-Resiga, R.F., et al., *Mathematical modelling of swirling flow in hydraulic turbines for the full operating range.* Applied Mathematical Modelling, 2011. **35**(10): p. 4759-4773.
15. T. Kubota, F.H., F. Avellan, *Performance analysis of the draft tube for GAMM Francis turbine.* Hydraulic Machinery and Cavitation, 1996. **1**: p. 130–139.
16. Oliveira Castro, A.P.F., Luciano Devinar, *The application of state of art CFD tools at the hydraulic turbine design*, in *24th Symposium on Hydraulic Machinery and Systems*. 2008: FOZ DO IGUASSU.
17. Holmes, G. and J.Y. McNabb, *Application of three-dimensional finite element potential flow analysis to hydraulic turbines*, in *International Symposium on Refined Modeling of Flows*. 1982: Paris, France.
18. Wu, J., et al., *CFD-Based Design Optimization for Hydro Turbines.* Journal of Fluids Engineering, 2007. **129**(2): p. 159-168.
19. *Fluent Software Training, TRN-99-003, Fluent Inc.* . 2001.
20. J.M. F-Nava, E.R.T., O.D. Gomez, J.M. F-Davila, R. R-Espinosa. *CFD performance evaluation and runner blades design optimization in a Francis turbine*. in *ASME 2009 Fluids Engineering Division Summer Meeting*. 2009. Colorado, USA.
21. Hu, Y., et al., *Numerical simulation of unsteady turbulent flow through a Francis turbine.* Wuhan University Journal of Natural Sciences, 2011. **16**(2): p. 179-184.
22. Yakhot, V. and S.A. Orszag, *Renormalization-Group Analysis of Turbulence.* Physical Review Letters, 1986. **57**(14): p. 1722-1724.

23. Yaras, M.I. and A.D. Grosvenor, *Evaluation of one- and two-equation low-Re turbulence models. Part I—Axisymmetric separating and swirling flows.* International Journal for Numerical Methods in Fluids, 2003. **42**(12): p. 1293-1319.
24. Chien, K.-Y., *Prediction of channel and boundary layer flows with a low-Reynolds-number turbulence model.* AIAA Journal, 1982. **20**: p. 33–38.
25. Rodi, W. *Recent developments in turbulence modeling*. in *3rd International Symposium on Refined Flow Modeling and Turbulence Measurements*. 1988. Tokyo, Japan,.
26. Wilcox, D., *Turbulence Modeling for CFD (2nd edn)*. 1988, DCW Industries: Canada.
27. Menter, F., *Eddy viscosity transport models and their relation to the k–e model*, in *NASA Technical Memorandum TM-108854*. 1994.
28. Spalart PR., A., SR., *A one equation turbulence model for aerodynamic flows.* AIAA Paper 92-0439, 1992.
29. Susan-Resiga, R., et al., *Analysis and Prevention of Vortex Breakdown in the Simplified Discharge Cone of a Francis Turbine.* Journal of Fluids Engineering, 2010. **132**(5): p. 051102-15.
30. Susan-Resiga, R., Muntean, S., Stein, P., and Avellan, F., *Axisymmetric Swirling Flow Simulation of the Draft Tube Vortex in Francis Turbines at Partial Discharge.* International Journal of Fluid Machinery and Systems, 2009. **2** (4): p. 295–302.
31. Shih, T.-H., et al., *A new k-ϵ eddy viscosity model for high reynolds number turbulent flows.* Computers & Fluids, 1995. **24**(3): p. 227-238.
32. Olivier, P., K. Chisachi, and A. François, *High-resolution LES of the rotating stall in a reduced scale model pump-turbine.* IOP Conference Series: Earth and Environmental Science, 2014. **22**(2): p. 022018.

33. Krappel, T., et al., *Validation of an IDDES-type turbulence model and application to a Francis pump turbine flow simulation in comparison with experimental results.* International Journal of Heat and Fluid Flow, 2015. **55**: p. 167-179.
34. Flores, E., et al., *Design of large Francis turbine using optimal methods.* IOP Conference Series: Earth and Environmental Science, 2012. **15**.
35. Martins, J.R.R.A., J.J. Alonso, and J.J. Reuther, *A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design.* Optimization and Engineering, 2005. **6**(1): p. 33-62.
36. Tatossian, C.A., S.K. Nadarajah, and P. Castonguay, *Aerodynamic shape optimization of hovering rotor blades using a Non-Linear Frequency Domain approach.* Computers & Fluids, 2011. **51**(1): p. 1-15.
37. Kolda, T.G., R.M. Lewis, and V. Torczon, *Optimization by Direct Search: New Perspectives on Some Classical and Modern Methods.* SIAM Review, 2003. **45**(3): p. 385-482.
38. Booker, A.J., et al., *A rigorous framework for optimization of expensive functions by surrogates.* Structural Optimization, 1999. **17**(1): p. 1-13.
39. Audet, C. and J. J. E. Dennis, *Analysis of Generalized Pattern Searches.* SIAM Journal on Optimization, 2002. **13**(3): p. 889-903.
40. Goldberg, D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*. 1989: Addison-Wesley.
41. Michalewicz, Z., *Genetic Algorithms + Data Structures = Evolution Programs*. 1996: Springer, Heidelberg.

42. Clerc, M. and J. Kennedy, *The particle swarm - explosion, stability, and convergence in a multidimensional complex space.* Evolutionary Computation, IEEE Transactions on, 2002. **6**(1): p. 58-73.

43. Kirkpatrick, S., C.D. Gelatt, and M.P. Vecchi, *Optimization by Simulated Annealing.* Science, 1983. **220**(4598): p. 671-680.

44. Ruan, H., et al., *Hydraulic design of a low-specific speed Francis runner for a hydraulic cooling tower.* IOP Conference Series: Earth and Environmental Science, 2012. **15**(3): p. 032011.

45. Shingai, K., et al. *Multi-objective optimization of diagonal flow type reversible pump-turbines*. in *24st IAHR Symposium on hydraulic machinery and systems*. 2010. Brazil.

46. Hare, W., J. Nutini, and S. Tesfamariam, *A survey of non-gradient optimization methods in structural engineering.* Advances in Engineering Software, 2013. **59**: p. 19-28.

47. Audet, C. and J. J. E. Dennis, *Mesh Adaptive Direct Search Algorithms for Constrained Optimization.* SIAM Journal on Optimization, 2006. **17**(1): p. 188-217.

48. Le Digabel, S., C. Tribes, and C. Audet, *NOMAD User Guide*. 2013.

49. Leifsson, L. and S. Koziel, *Variable-Fidelity Aerodynamic Shape Optimization*, in *Computational Optimization, Methods and Algorithms*, S. Koziel and X.-S. Yang, Editors. 2011, Springer Berlin Heidelberg. p. 179-210.

50. Audet, C., V. Béchard, and S. Le Digabel, *Nonsmooth optimization through Mesh Adaptive Direct Search and Variable Neighborhood Search.* Journal of Global Optimization, 2008. **41**(2): p. 299-318.

51. Audet, C., G. Savard, and W. Zghal, *Multiobjective Optimization Through a Series of Single-Objective Formulations.* SIAM Journal on Optimization, 2008. **19**(1): p. 22.

52. Audet, C., G. Savard, and W. Zghal, *A mesh adaptive direct search algorithm for multiobjective optimization.* European Journal of Operational Research, 2010. **204**(3): p. 545-556.
53. Leifsson, L. and S. Koziel, *Multi-fidelity design optimization of transonic airfoils using physics-based surrogate modeling and shape-preserving response prediction.* Journal of Computational Science, 2010. **1**(2): p. 98-106.
54. Friedberg, R.M., *A learning machine: part I.* IBM J. Res. Dev., 1958. **2**(1): p. 2-13.
55. Bremermann, H.J., *Optimization through evolution and recombination*. 1962.
56. Box, G.E.P., *Evolutionary Operation: A Method for Increasing Industrial Productivity.* Journal of the Royal Statistical Society. Series C (Applied Statistics), 1957. **6**(2): p. 81-101.
57. Kyriacou, S., *Evolutionary Algorithm-based Design-Optimization Methods in Turbomachinery*. 2013, National Technical University of Athens.
58. Deb, K., *An efficient constraint handling method for genetic algorithms.* Computer Methods in Applied Mechanics and Engineering, 2000. **186**(2–4): p. 311-338.
59. Giannakoglou, K.C., *Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence.* Progress in Aerospace Sciences, 2002. **38**(1): p. 43-76.
60. Forrester, A.I.J. and A.J. Keane, *Recent advances in surrogate-based optimization.* Progress in Aerospace Sciences, 2009. **45**(1–3): p. 50-79.
61. Koziel, S., D. Ciaurri, and L. Leifsson, *Surrogate-Based Methods, Computational Optimization, Methods and Algorithms*, S. Koziel and X.-S. Yang, Editors. 2011, Springer Berlin / Heidelberg. p. 33-59.
62. Box, G.E.P.D., Norman R. , *Empirical model-building and response surfaces. Wiley series in probability and mathematical statistics*. 1987, Oxford, England: John Wiley & Sons.

63. H.A. Georgopoulou, S.A.K., K.C. Giannakoglou, P. Grafenberger, E. Parkinson. *Constrained Multi-Objective Design Optimization of Hydraulic Components Using a Hierarchical Metamodel Assisted Evolutionary Algorithm. Part 1: Theory*. in *24th Symposium on Hydraulic Machinery and Systems*. 2008.

64. Karakasis, M.K. and K.C. Giannakoglou, *On the use of metamodel-assisted, multi-objective evolutionary algorithms.* Engineering Optimization, 2006. **38**(8): p. 941-957.

65. Jones, D.R., M. Schonlau, and W.J. Welch, *Efficient Global Optimization of Expensive Black-Box Functions.* Journal of Global Optimization, 1998. **13**(4): p. 455-492.

66. Jouhaud, J.C., P. Sagaut, and M. Montagnac, *A surrogate-model based multidisciplinary shape optimization method with application to a 2D subsonic airfoil.* Computers & Fluids, 2007. **36**(3): p. 520-529.

67. Badhurshah, R. and A. Samad, *Multiple surrogate based optimization of a bidirectional impulse turbine for wave energy conversion.* Renewable Energy, 2015. **74**(0): p. 749-760.

68. Vesting, F. and R.E. Bensow, *On surrogate methods in propeller optimisation.* Ocean Engineering, 2014. **88**(0): p. 214-227.

69. J. Reuther, A.J., *Control Theory Based Airfoil Design for Potential Flow and a Finitie Volume Discretization*, in *AIAA 32nd Aerospace Sciences Meeting and Exhibit*. 1994: Reno, Nevada.

70. A. Jameson, J.R. *Control Theory Based Airfoil Design using Euler Equations,*. in *AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. 1994. Panama City Beach.

71. Forrester, A.I.J., N.W. Bressloff, and A.J. Keane, *Optimization using surrogate models and partially converged computational fluid dynamics simulations.* Proceedings of the Royal

Society of London, Series A (Mathematical, Physical and Engineering Sciences), 2006. **462**(2071): p. 2177-204.

72. Leary, S.J., A. Bhaskar, and A.J. Keane, *A knowledge-based approach to response surface modelling in multifidelity optimization.* Journal of Global Optimization, 2003. **26**(3): p. 297-319.
73. Alexandrov, N. and R. Lewis, *An Overview of First-Order Model Management for Engineering Optimization.* Optimization and Engineering, 2001. **2**(4): p. 413-430.
74. Conn, A., N. Gould, and P. Toint, *Trust-Region Methods (MPS-SIAM Series on Optimization)*. 2000: Society for Industrial and Applied Mathematics.
75. Alexandrov, N., et al., *First-order model management with variable-fidelity physics applied to multi-element airfoil optimization*, in *8th Symposium on Multidisciplinary Analysis and Optimization*. 2000, American Institute of Aeronautics and Astronautics.
76. Kolencherry, N.J., *Multi-fidelity optimization strategies using genetic algorithms and sequential kriging surrogates*. 2011, Purdue University.
77. Rao, S.S., *Engineering Optimization: Theory and Practice*. 3rd ed. 1996: Wiley.
78. Bandler, J.W., et al., *Space mapping: the state of the art.* IEEE Transactions on Microwave Theory and Techniques, 2004. **52**(1): p. 337-61.
79. Koziel, S., J.W. Bandler, and K. Madsen, *A Space-Mapping Framework for Engineering Optimization—Theory and Implementation.* Microwave Theory and Techniques, IEEE Transactions on, 2006. **54**(10): p. 3721-3730.
80. Koziel, S., Q.S. Cheng, and J.W. Bandler, *Space mapping.* Microwave Magazine, IEEE, 2008. **9**(6): p. 105-122.

81. Echeverría, D., D. Lahaye, and P. Hemker, *Space Mapping and Defect Correction*, in *Model Order Reduction: Theory, Research Aspects and Applications*, W.A. Schilders, H. van der Vorst, and J. Rommes, Editors. 2008, Springer Berlin Heidelberg. p. 157-176.
82. Serafini, B., *A framework for managing models in nonlinear optimization of computationally expensive functions*. 1998, Rice University: Houston.
83. Giannakoglou, K.C., D.I. Papadimitriou, and I.C. Kampolis, *Aerodynamic shape design using evolutionary algorithms and new gradient-assisted metamodels.* Computer Methods in Applied Mechanics and Engineering, 2006. **195**(44-47): p. 6312-29.
84. Mengistu, T. and W. Ghaly, *Aerodynamic optimization of turbomachinery blades using evolutionary methods and ANN-based surrogate models.* Optimization and Engineering, 2008. **9**(3): p. 239-255.
85. Kuahai, Y., Y. Xi, and Y. Zhufeng, *Aerodynamic and heat transfer design optimization of internally cooling turbine blade based different surrogate models.* Structural and Multidisciplinary Optimization, 2011. **44**(1): p. 75-83.
86. Jameson, A. and J. Reuther, *Control Theory Based Airfoil Design using Euler Equations*, in *AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. 1994: Panama City Beach.
87. Bandler, J.W., et al., *Space mapping: the state of the art.* Microwave Theory and Techniques, IEEE Transactions on, 2004. **52**(1): p. 337-361.
88. Robinson, T.D., et al., *Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping.* AIAA Journal, 2008. **46**(11): p. 2814-2822.
89. Alexandrov, N.M., et al., *Optimization With Variable-Fidelity Models Applied to Wing Design*. 2000, NASA Langley Technical Report Server.

90. Koziel, S. *Efficient optimization of microwave circuits using shape-preserving response prediction*. in *Microwave Symposium Digest, 2009. MTT '09. IEEE MTT-S International*. 2009.

91. Wu, J., et al., *CFD-Based Design Optimization for Hydro Turbines.* Journal of Fluids Engineering, 2006. **129**(2): p. 159-168.

92. Bahrami, S., et al. *Multi-objective optimization of runner blades using a multi-fidelity algorithm*. in *ASME 2013 Power Conference*. 2013. Boston, USA.

93. Le Digabel, S., *Algorithm 909: NOMAD: nonlinear optimization with the MADS algorithm.* ACM Transactions on Mathematical Software, 2011. **37**(4): p. 44 (15 pp.).

94. Abramson, M., et al. *The NOMAD project*. Available from: http://www.gerad.ca/nomad.

95. Abramson, M.A., et al., *ORTHOMADS: a deterministic MADS instance with orthogonal directions.* SIAM Journal on Optimization, 2009. **20**(2): p. 948-66.

96. Audet, C. and J.E. Dennis, Jr., *A progressive barrier for derivative-free nonlinear programming.* SIAM Journal on Optimization, 2009. **20**(1): p. 445-72.

97. Gheribi, A.E., et al., *Identifying optimal conditions for magnesium based alloy design using the Mesh Adaptive Direct Search algorithm.* Thermochimica Acta, 2013. **559**(0): p. 107-110.

98. Audet, C., et al., *Metamaterial Design by Mesh Adaptive Direct Search*, in *Numerical Methods for Metamaterial Design*, K. Diest, Editor. 2013, Springer Netherlands. p. 71-96.

99. Hare, W., J. Nutini, and S. Tesfamariam, *A survey of non-gradient optimization methods in structural engineering.* Advances in Engineering Software, 2013. **59**(0): p. 19-28.

100. Yang, X.-S., *Introduction to Mathematical Optimization*. 2008: Cambridge International Science Publishing.

101. Postaire, J.G., R.D. Zhang, and C. Lecocq-Botte, *Cluster analysis by binary morphology*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1993. **15**(2): p. 170-80.

102. Dominique, S., J.Y. Trépanier, and C. Tribes, *GATE: a genetic algorithm designed for expensive cost functions.* International Journal of Mathematical Modelling and Numerical Optimisation, 2012. **3**: p. 5-29.

103. Nilsson, H. and L. Davidson, *Validations of CFD against detailed velocity and pressure measurements in water turbine runner flow.* International Journal of Numerical Methods in Fluids, 2003. **41**(8): p. 16.

104. Zhang, M.-L., C.W. Li, and Y.-M. Shen, *A 3D non-linear k–ε turbulent model for prediction of flow and mass transport in channel with vegetation.* Applied Mathematical Modelling, 2010. **34**(4): p. 1021-1031.

105. Gauthier, M., *StageX Package*. 2012, Andritz Hydro Ltd.

106. Vu, T.C., et al., *Steady and unsteady flow computation in an elbow draft tube with experimental validation.* International Journal of Fluid Machinery and Systems, 2010. **4**(1): p. 84-95.

107. Vu, T.C., et al. *Flow simulation for a propeller turbine with different runner blade geometries*. in *26th IAHR Symposium on Hydraulic Machinery and Systems, August 19, 2012 - August 23, 2012*. 2012. Beijing, China: Institute of Physics Publishing.

108. Susan-Resiga, R., et al., *Axisymmetric Swirling Flow Simulation of the Draft Tube Vortex in Francis Turbines at Partial Discharge*, in *24th Symposium on Hydraulic Machinery and Systems*. 2008: Brazil.

109. Murry, N., *Xmt Overview*. 2010, Andritz Hydro Ltd.

110. Franco-Nava, J.M., et al., *CFD performance evaluation and runner blades design optimization in a Francis turbine*, in *FEDSM 2009, ASME 2009 Fluids Engineering Division Summer Meeting*. 2009: Colorado, USA.

111. Pilev, I.M., et al., *Multiobjective optimal design of runner blade using efficiency and draft tube pulsation criteria.* IOP Conference Series: Earth and Environmental Science, 2012. **15**(3): p. 032003.

112. Derakhshan, S. and N. Kasaeian, *Optimal design of axial hydro turbine for micro hydropower plants.* IOP Conference Series: Earth and Environmental Science, 2012. **15**(4): p. 042029.

113. Bahrami, S., et al., *Multi-fidelity design optimization of Francis turbine runner blades.* IOP Conference Series: Earth and Environmental Science, 2014. **22**(1): p. 012029.

114. Giannakoglou, K.C. *The EASY (Evolutionary Algorithms SYstem) software*. 2008; Available from: http://velos0.ltt.mech.ntua.gr/EASY.

115. Kyriacou, S., et al., *Evolutionary algorithm based optimization of hydraulic machines utilizing a state-of-the-art block coupled CFD solver and parametric geometry and mesh generation tools.* IOP Conf. Ser.: Earth Environ. Sci., 2014. **22**(012024).

116. McNabb, J., et al., *CFD based draft tube hydraulic design optimization.* IOP Conf. Ser.: Earth Environ. Sci., 2014. **22**(012023).

117. Bahrami, S., et al., *Multi-fidelity shape optimization of hydraulic turbine runner blades using a multi-objective mesh adaptive direct search algorithm.* Applied mathematical modelling, 2016. **40**: p. 18.

118. Fletcher, R. and S. Leyffer, *Nonlinear programming without a penalty function.* Mathematical Programming, 2002. **91**(2): p. 239-269.

119. Gutmann, H.M., *A Radial Basis Function Method for Global Optimization.* Journal of Global Optimization, 2001. **19**(3): p. 201-227.

120. Jones, D.R., *A taxonomy of global optimization methods based on response surfaces.* Journal of Global Optimization, 2001. **21**(4): p. 345-383.

121. Oyama, A., M.-S. Liou, and S. Obayashi, *Transonic Axial-Flow Blade Optimization: Evolutionary Algorithms/Three-Dimensional Navier-Stokes Solver*. Journal of Propulsion and Power, 2004. **20**(4): p. 612-619.

122. Pierret, S., R. Filomeno Coelho, and H. Kato, *Multidisciplinary and multiple operating points shape optimization of three-dimensional compressor blades.* Structural and Multidisciplinary Optimization, 2007. **33**(1): p. 61-70.

123. Strazisar, A.J., et al., *Laser anemometer measurements in a transonic axial-flow fan rotor*. 1989, NASA Lewis Research Center.

124. Lupien, A., *Une approche 3D pour la paramétrisation de pales de turbosoufflantes transsoniques*, in *Mechanical engineering department*. 2011, École polytechnique de Montréal: Montréal.

125. Khelghatibana, M., *An Approach for Aerodynamic Optimizaton of Transonic Fan Blades*, in *Mechanical engineering department*. 2014, École Polytechnique de Montréal.

126. Khelghatibana, M., et al. *Multi-objective and multi-point aerodynamic optimization of transonic fan blades*. in *ASME 2014 International Mechanical Engineering Congress and Exposition*. 2014. Montreal, Canada.

127. Luo, J., J. Xiong, and F. Liu, *Aerodynamic design optimization by using a continuous adjoint method.* Science China Physics, Mechanics & Astronomy, 2014. **57**(7): p. 1363-1375.

128. Deb, K., *Multi-objective optimization using evolutionary algorithms*. 2001: John Wiley & Sons Chichester.

129. Akira, O., L. Meng-Sing, and O. Shigeru, *Transonic Axial-Flow Blade Shape Optimization Using Evolutionary Algorithm and Three-Dimensional Navier-Stoke Solver*, in *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. 2002, American Institute of Aeronautics and Astronautics.